University of Alberta

Laser-Field Femtosecond Electron Pulse Generation using Surface Plasmons

by

Scott Edward Irvine

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Electrical and Computer Engineering

Edmonton, Alberta
Spring 2007

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-29692-9
Our file Notre référence
ISBN: 978-0-494-29692-9

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

Abstract

This thesis presents experimental and theoretical work on the generation of energetic electron bunches using surface plasmon (SP) waves. High-intensity laser pulses are used to excite these electromagnetic charge oscillations at a metal-vacuum interface to ponderomotively accelerate electrons to moderate energies.

Initial experiments focus on the excitation of SP waves using low-energy pulses from laser oscillator. For the first time, it is shown that electrons with energies ranging up to 0.4 keV could be generated using only 1.5 nJ pulses. Further experiments using higher energy pulses (0.5 mJ) from a laser amplifier indicate that electrons with energies ranging up to 2 keV can be generated. A detailed examination of the photo-ejection process reveals that electron emission characteristics depart from multiphoton absorption towards Keldysh-like field ionization for power densities above ~2 GW/cm^2.

In addition to the experimental endeavours, a novel quasi-classical model describing SP electron acceleration is derived. The model is based on finite-difference time-domain solution of Maxwell's equations and includes the nonlinear electron photoemission characteristics of metallic surfaces. This model provides unique visualization of the acceleration process, and reveals a new bunching mechanism for the photo-accelerated electron pulse. Excellent agreement between the model energy spectra and the experimentally measured electron energy distributions is obtained. The agreement between the measured and simulated spectra indicates that the electric field of the laser is

enhanced by over three orders of magnitude. Furthermore, comparison of the experimental and calculated electron energy spectra shows that surface roughness plays an important role in the enhancement of the electric field.

Two novel physical processes surrounding SP electron acceleration have also been explored using the developed model. First, a method for electron beam slicing is proposed. Analysis indicates that a large portion of an incident electron beam can be time-gated with a precision limited only by the laser pulse (~30 fs) and can be utilized for temporal characterization of ultrashort electron bunches below 100 fs. Second, it is demonstrated that SP electron acceleration can be coherently controlled through the carrier-envelope phase (CEP) of the excitation optical pulse. The physical origin of the CEP-sensitivity arises from an electron's ponderomotive interaction with the oscillating electromagnetic field of the SP wave.

Acknowledgements

First of all I would like to thank my supervisor, Abdulhakem Y. Elezzabi, for his support, motivation, and guidance during the launching of my career. Also, a thank you to Blair Harwood, Herbert Dexel, Barry Arnold, Martin Reidner, Mike Boissonneault, and Reiner Schwarze for their excellent technical advice over the past four years. I would also like to thank my peers Jonathan F. Holzman, Kenneth J. Chau, and Vikram Kohli for their camaraderie.

Equally important are the members of my family, who have given me everything: this thesis is truly dedicated to them. Most important, is the LOVE and continued support that my wife, Angela, has given me during my graduate studies.

Table of Contents

1 Introduction 1

1.1 Methods for Generating Electron Pulses 3

1.2 Surface Plasmons 5

1.3 Thesis Objectives 9

1.4 Applications 10

1.5 Thesis Organization 13

1.6 References 14

2 Background 20

2.1 Drude Theory of Metals 21

2.2 Surface Plasmon Waves 24

2.3 Electromagnetic Wave Coupling to Surface Plasmons 28

2.4 Electron Emission from Metallic Surfaces 35

2.5 The Ponderomotive Force 38

2.6 Literature Review 41

2.7 Thesis Goals 43

2.8 References 44

3 Theoretical Model 47

3.1 Finite-Difference Time-Domain Numerical Method 49

3.2 Electron Photoemission and Motion 59

3.3 Model Results 61

3.4 Summary 77

3.5 References 79

4 Experimental Results....81

4.1 Experiments Using Low-Energy High-Repetition Rate Femtosecond Pulses from a Laser Oscillator....82

4.2 Experiments Using High-Energy Femtosecond Pulses from a Laser Amplifier....92

4.3 Comparison of Model Calculations with Experimental Measurements....104

4.4 Summary....112

4.5 References....113

5 Further Model Predictions....116

5.1 Electron Beam Slicing and Gating....118

5.1.1 Geometry for Electron Beam Slicing....118

5.1.2 Model Results for Electron Beam Slicing....122

5.2 Influence of Carrier-Envelope Phase on SP Electron Acceleration....140

5.3 Summary....149

5.4 References....151

6 Conclusion....153

6.1 Future Directions....155

6.2 References....168

Appendix A Magnetic Spectrometer Design....170

A.1 Introduction....171

A.2 Device Schematic and Operational Principles....172

A.3 Results and Discussion....177

A.3.1 Magnetic Field Distribution and Trajectory Analysis.....177

A.3.2 Electron Multiplier Response.....180

A.4 References.....183

Appendix B Model Implementation and Source Code.....186

B.1 Program Code.....189

Curriculum Vitae 228

List of Tables

Table B.1. Various classes used in the simulation and a brief description of their function. 188

Table B.2. Various classes used in the simulation and associated files. 189

List of Figures

Figure 1.1. Illustration of the acceleration grid method used to generate energetic electron pulses for conventional time-resolved electron diffraction systems. 4

Figure 1.2. A graphical illustration of ponderomotive electron acceleration using surface plasmon waves. 9

Figure 2.1. Optical constants (a) and dielectric function (b) as calculated from the Drude model. 23

Figure 2.2. Two dielectric half-spaces which are divided at $z = 0$. At the interface, an SP oscillation, propagating along the positive x-direction, can exist. 24

Figure 2.3. (solid line) Dispersion curve of a surface plasmon oscillation. Photon dispersion curves are also shown (dashed lines) for the cases of free-space mode and a wave traveling in a dielectric medium. The axes are normalized to the SP frequency and wavevector. 29

Figure 2.4. Prism geometry for coupling free-space electromagnetic waves to surface plasmon oscillations, also known as the Kretschmann configuration. 30

Figure 2.5. (top) Three-layer system comprised of vacuum-metal-dielectric media. Portions of an incident beam experience multiple reflections within the metallic film, which must be summed to calculate the effective Fresnel coefficient of reflection, r_{210}, for the entire structure. (bottom) Reflection coefficient of the multi-layer structure as a function of angle, clearly illustrating resonance absorption of the incident beam and coupling to SP waves. 32

Figure 2.6. Comparison of linear (left) and multiphoton (right) electron emission from a metallic surface. For any single electronic transition, the probability of electron excitation is linearly proportional to the intensity of the exciting laser radiation. When multiple (virtual) transitions are involved, the scaling relationship is an integer power that is equal to the number of virtual transitions. 36

Figure 2.7. Illustration of the process of electron acceleration via the ponderomotive force. Electrons, when subjected to a spatial electric field gradient, will feel asymmetric forces in the subsequent oscillations of the wave. This results in a net gain in velocity in the direction of decreasing electric field/amplitude. In the case of a surface plasmon, the direction is away from the metal film surface. 38

Figure 2.8. Prism coupling method used by Zawadzka et al. [11,12] to launch surface plasmons and accelerate electrons. 41

Figure 2.9. Grating coupling method used by Kupersztych et al. [13] to launch surface plasmons and accelerate electrons. 42

Figure 3.1. (top) Spatial cells of the electric and magnetic field discretization grids, illustrating the relative locations of the lattice points where the E_x, E_y, and H_z field components are calculated. The displacement vector components D_x and D_y share the same points as the corresponding electric field components, however, are omitted for clarity. (bottom) Illustration of the discretization grids along the temporal dimension, indicating that the electric and magnetic fields are also 'leapfrogged' in time as well as in space. 52

Figure 3.2. (a) Computational domain used for the FDTD portion of the model. The electromagnetic (EM) wave is initiated using the total-field scattered-field (TF SF)

technique, and perfectly matched layers (PML) are used to eliminate non-physical reflections from the computational boundaries. (b) PML system used to provide the zero reflection, highly absorbing material to prevent electromagnetic wave reflection at the outermost boundaries of the computational window. 55

Figure 3.3. (a) Distribution of the total electric field during the excitation of the SP wave. An enhanced electric field is observed at the surface which decays into vacuum with a characteristic length of α_0^{-1}=240 nm. (b) Vector representation of the electric field distribution near the silver film surface, which is overlaid with the local charge density. ... .63

Figure 3.4. A sample set of 20 electron trajectories are shown. The quivering motion of the electrons in the oscillating electromagnetic field of the SP is evident with a distance of ~36 nm between the inflection points of the most energetic electrons. 64

Figure 3.5. Kinetic energy distribution of the SP accelerated electrons for E_{SP} values of 1.9×10^9, 2.7×10^9, and 3.7×10^9 V/cm. 65

Figure 3.6. Calculated variation of maximum observed kinetic energy with E_{SP} (circles) for a τ_p=30 fs pulse. The linear fit (solid line) indicates a 2.6 power dependence. 67

Figure 3.7. A simple scaling relation between maximum kinetic energy and electric field for electrons ponderomotively accelerated with an ultrashort pulse with duration of τ_p=30 fs and α_0^{-1}=240 nm. For electric fields below 4.0×10^8 V/cm, a fourth order power dependence is observed while a second order power dependence exists at higher field strengths. 69

Figure 3.8. Variation of maximum observed kinetic energy with E_{SP} for a ~1 ps excitation pulse, calculated from the FDTD model, indicating a 2.4 power dependence. 70

Figure 3.9. Snap-shots at times of -20, 20, 60, 100, 140, and 180 fs illustrate the behavior of electrons under the influence of the SP field. Time zero indicates when the center of the excitation pulse reaches the silver film. The marker in the final frame (180 fs) designates the location of electron number distribution sampling in relation to the metal surface (dashed line). The results of the electron number distribution sampling are shown in Figure 3.10. 72

Figure 3.10. Sampled electron number distribution as a function of time for a fixed location ~140 nm in front of the silver film surface (indicated in Figure 3.9 by a cross-hair). The effect of microbunching is evident as several individual ~ 4 fs peaks have developed within the envelope. 74

Figure 3.11. In-plane angular distribution of the photo-accelerated electrons. The peak emission angle occurs at 18.2° away from the normal to the surface. 75

Figure 3.12. Angle-resolved energy distributions for several E_{SP} of: a) 3.7×10^8, b) 7.4×10^8, c) 1.9×10^9, d) 2.7×10^9, e) 3.7×10^9, and f) 7.4×10^9 V/cm. 76

Figure 3.13. Selected slices along three panels of Figure 3.12 corresponding to a) 3.7×10^8, b) 2.7×10^9, and c) 7.4×10^9 V/cm. These energy distributions demonstrate the selectivity and control over the shape and maximum energy of the spectra. 78

Figure 4.1. Experimental arrangement for SP electron acceleration. HVC: high vacuum chamber, Ti:S: Kerr-lens mode-locked titanium-sapphire laser oscillator, RPA: retarding

potential analyzer. Energy spectra are determined by tracking the amplified photocurrent i_{photo} as a function of the grid voltage V_{bias}. 83

Figure 4.2. (a) Measured photocurrent dependence vs. intensity verifying a three-photon process for 800 nm laser oscillator pulses irradiating a silver metal film in the Kretschmann configuration. (b) Measured photocurrent dependence vs. intensity using 5fs laser pulses from an ultra-broadband titanium-sapphire oscillator. 85

Figure 4.3. (a) Measured interferometric autocorrelation trace using 5 fs laser pulses from an ultra-broadband titanium-sapphire oscillator. (b) Calculated autocorrelation trace using equation 4.1 and $\tau_p = 5$ fs optical pulses. Comparison between (a) and (b) indicates that no broadening has occurred. 88

Figure 4.4. (a) Typical electrical signals obtained using the retarding potential analyzer using 1.5 nJ optical pulses from the titanium-sapphire laser oscillator at an intensity of 0.5 GW/cm^2. Each curve represents a different location on the prism surface. (b) Selected integrated spectrum illustrating that electrons with energies up to 0.4 keV are present. (c) Experimental kinetic energy distribution of the femtosecond electron pulses obtained by differentiating the data in panel (b). 90

Figure 4.5. (solid line) Measured reflectance of the optical beam in the Kretschmann configuration using the laser amplifier system and a gold metal film. The resonant absorption dip clearly indicates coupling to SP modes. The resonance angle occurs at a value of $\theta_{SP} = 41.7°$ (vertical dotted line) and the full-width at half-maximum is 4°. Also shown is the electron emission as a function of angle of incidence of the laser beam (dashed line), which indicates that photoemission of electrons is directly correlated with SP coupling. 93

Figure 4.6. (left) Apparatus used to measure the angular distribution of the photo-accelerated electrons. A rack and pinion drive system is implemented to sweep a Faraday cup about the prism surface. The resulting spectra are plotted for both (a) silver and (b) gold. Peak emission occurs near 10° and the distribution are nearly symmetric about this direction with angular half-widths of ~50°. 94

Figure 4.7. Calculated and measured interferometric autocorrelation traces for both (a) silver and (b) gold surfaces. The traces reveal that the optical pulse has broadened to 100 fs during its propagation through the prism. More striking is the fact that the orders of the autocorrelations are 1.75 for silver and 1.65 for gold; much less than the expected 3rd and 4th order dependencies. 96

Figure 4.8. Dependence of photocurrent on pump intensity for (a) silver and (b) gold. For lower intensities, 3rd and 4th order power dependencies are observed. However, at higher intensities the orders are reduced in both metals to 1.47 and 1.26 for silver and gold, respectively. (c) A similar dependence of photocurrent on pump intensity is obtained for a silver film that has been exposed to atmosphere for ~24 hours. In this case the multiphoton order is increase, as is the order in the tunnel regime. 97

Figure 4.9. (a) Time-of-flight electron energy spectrometer. Electrons, emitted from the prism surface, travel the length of a 0.38 m flight tube and are detected by a microchannel plate (MCP) array. The signal from the MCP is processed using a 20 GHz oscilloscope (Tektronix CSA8000, 17.5 ps rise-time module). (b) The solid curve is the electrical waveform measured with the time-of-flight apparatus (close-up view of highest energy electrons). A dashed curve is also shown and corresponds to laser light that is sent directly through the apparatus without coupling to SP waves. (c) Signal produced from

the difference of the waveforms shown in panel (b). The vertical lines in panels (b) and (c) indicate the zero time, which represents the time at which the electrons are emitted from the prism surface. (d) Kinetic energy spectrum of the photo-accelerated electrons determined from the waveform in (c) and a zero time of 95.9 ns. 100

Figure 4.10. Kinetic energy spectrum measured using the magnetic sector analyzer described in Appendix A. Error bars show the reduced sensitivity at lower signal levels. The inset illustrates the configuration of the device, which included a 10° wedge to improve the signal yield (see Figure 4.6). To achieve a device resolution of 10%, two apertures (not shown) are placed at the electron source (gold-coated prism) and detector and have widths of 2 mm and 1 mm, respectively. An additional 1 mm aperture is placed mid-way through the central chamber arc. 103

Figure 4.11. Integrated kinetic energy spectrum measured using the retarding potential method shown in Figure 4.1. A broad range of spectral components are present from 0 to 2 keV for both silver (circle) and gold (square) surfaces. The error is indicated only on the final point of the silver surface curve for clarity and applies to all data points. 105

Figure 4.12. Comparison between measured electron energy spectra using the laser oscillator (circles) and theoretical energy spectra (solid) as calculated from the model described in Chapter 3. 106

Figure 4.13. Calculated (solid lines) and experimentally measured (circles) integral kinetic energy distributions for (a) silver and (b) gold metal films. The calculated (solid lines) and experimentally measured (circles) in-plane angular distributions are shown in the corresponding insets. 108

Figure 4.14. (a) Graphic illustration comparing the focusing conditions for the laser amplifier and laser oscillator experiments. For the amplifier, the entire beam is coupled to SP waves as each portion of the beam has the same propagation vector. In contrast, the oscillator beam is focused, and therefore, only rays near the optical axis have the necessary momentum to couple to SPs. The image next the depiction of the oscillator illustrates a measurement taken using a focused helium-neon laser beam. It is clear that only the central portion (6%) is efficiently coupled. Panels (b) and (d) show atomic force microscope images of two silver surfaces deposited by (b) magnetron and (d) ion-beam sputtering techniques. The vertical scale (gray scale) on the right indicates that the surface features of (b) have heights up to 40 nm. Graph (c) is a two-dimensional FDTD simulation of the electric field distribution in the vicinity of a surface defect (25 nm pyramid). The additional enhancement due to confinement is clear ($\eta \sim 12$). 111

Figure 5.1. Arrangement for electron pulse gating using SP waves. (top) An external electron beam is directed toward a metal-coated prism surface at angle θ, measured from the surface normal. The electrons comprising the input beam are deflected and depart the interaction region at an angle α. Varying the delay between the launching of the SP and an incident electron packet allows the same SP-gating mechanism to be utilized for temporal characterization of electron pulses. (bottom) Potential experimental arrangement for realizing electron beam gating using SP waves, which consists of a laser source, an electron source, a timing mechanism to synchronize the optical and electron pulses, and an electron spectrometer for energy discrimination. 119

Figure 5.2. Trajectories of five test electrons as they interact with an SP wave having a peak electric field amplitude of E_{SP}=7.4×10^9 V/cm. The test electrons are delayed with respect to the peak of E_{SP} with launching times of τ_e = -12, -6, 0, 6 and 12 fs. 123

Figure 5.3. Electrons interacting with the SP wave for various θ of (a) 75°, (b) 60°, (c) 45°, (d) 30°, (e) 15°, and (f) 0°. For each panel, two electron trajectories are plotted corresponding to the cases of $K_\perp < U_{SP}$ (solid blue) and $K_\perp > U_{SP}$ (dashed red). The arrows indicate the direction of the electrons as they approach and exit E_{SP}. 126

Figure 5.4. Comparison of threshold values of the ponderomotive potential required for electron deflection as calculated from the model (circles) and equation 8 (solid line). 128

Figure 5.5. Snapshots of the SP-gating of an electron beam at various times ranging from -20 fs to 130 fs. The white arrow indicates the direction of the wavevector of the incident laser pulse, while the black arrow indicates the propagation direction of the electron beam. 130

Figure 5.6. (a) Number of deflected electrons as a function of time at various sample locations at perpendicular distances of 1.0, 1.5, and 2.0 μm away from the metal film surface. (b) Variation of the FWHM of the electron packet as a function of distance away from the prism surface along a direction 22° from the surface normal. (c) Variation of the total number of deflected electrons as a function of distance away from the prism surface along a direction 22° away from the surface normal. (d) Number of deflected electrons as a function of time at two locations along the film surface for distances of 3.0 and 4.0 μm away from the center of the metal surface. Legends in (a) and (d) contain illustrations

depicting the location of the detector with respect to the prism surface. It should be noted that in (a) and (d) the curves have been offset vertically for clarity. 132

Figure 5.7. (a) Kinetic energy spectra of the sliced electron beam depicted in Figure 5.6. The peak has a central value of 330 eV and a FWHM of 179 eV. The inset shows the angular distribution of the sliced electron beam and its relation to the surface of the prism. Directionality of the sliced beam is evidenced by the peak at 22°, which has an angular half-width of 21°. (b) Angle-resolved energy spectra reveal distinct energy bands that follow $K \sim K_{\parallel}\left(1+\cot^2\alpha\right)$ for various $K_{\parallel}$ values ranging from 204 to 417 eV. 134

Figure 5.8. Snapshots of the SP-gating of an electron pulse at various times ranging from -20 fs to 130 fs for a relative delay of $\tau = 0$. The arrows indicate the direction of the propagation of the electron and optical pulses. 136

Figure 5.9. Cross-correlation between an SP excited with a 30 fs optical pulse and a 50 fs electron pulse for various E_{SP} of 3.7×10^9 V/cm, 7.4×10^9 V/cm, and 1.9×10^{10} V/cm. ..137

Figure 5.10. Cross-correlation between an SP excited with a 30 fs optical pulse and electron pulses with durations ranging from 5 to 200 fs. The resolution of the system determined from the 5 fs electron pulse is 77 fs. 139

Figure 5.11. Upper left panel: illustration of the launching of an SP wave and subsequent dynamics of photo-injected electrons accelerated during the interaction with an SP wave excited with a τ_{laser}=5 fs. The succeeding panels depict the electron energy spectra at various φ_{CEP} ranging from 0 to 2π. For each value of φ_{CEP}, the insets show the electric field waveform, $E_L\left(t,\varphi_{CEP}\right) = E_0(t)\cos\left(\omega t+\varphi_{CEP}\right)$, and its specific relation to the Gaussian optical pulse envelope. Two pronounced cutoffs, positioned at values of $\delta_1 = 425$ eV and $\delta_2 = 685$ eV, are evident within the energy spectra. 141

Figure 5.12. (a) Overlapped energy spectra of SP-accelerated electrons for φ_{CEP} ranging from 0 to 2π and $\tau_p = 5$ fs. (b) The variation of the total number of electrons above $K_C = 300$ eV, which is also indicated by a dashed line in (a). 144

Figure 5.13. (a) Overlapped energy spectra of SP-accelerated electrons for φ_{CEP} ranging from 0 to 2π and $\tau_p = 12$ fs. The arrows in (a) indicate regions of CEP sensitivity. (b) The variation of the total number of electrons above $K_C = 720$ eV, which is also indicated by a dashed line in (a). Panel (c) illustrates overlapped energy spectra for $\tau_p = 30$ fs, which shows no indication of CEP effects. 146

Figure 5.14. $\Delta Q(K_C, \varphi_{CEP})$ surface plots illustrating the electron count as a function of both K_C and φ_{CEP} for (a) $\tau_p = 5$ fs and (b) $\tau_p = 12$ fs. Constant K_C cross-sections along surfaces are shown for both $\tau_p = 5$ fs and 12 fs, indicating that $Q(K_C, \varphi_{CEP})$ can be tailored to yield either 'sine-like' or 'cosine-like' waveforms. 148

Figure 6.1. Depiction of a phase-meter: a device capable of determining the absolute carrier-envelope phase of a few-cycle laser pulse. 157

Figure 6.2. The first two panels illustrate atomic force microscope measurements, repeated from Chapter 4, of two silver surfaces that have been deposited using two different methods of metallization: (a) magnetron sputtering and (b) ion-beam sputtering. The next four panels illustrate other types of nano-structures that hold promise for research of electron acceleration and electron pulse dynamics. Panel (c) shows nano-particles created by pulsed laser deposition. Periodic nano-pyramids, depicted in (d), can be created through anisotropic etching of a silicon surface. Other geometries that can be investigated include the (e) grating method for launching surface plasmons as well as (f)

the holey fiber. In the latter case, the fiber tip can be either coated with a metal film or impregnated with metallic nano-particles. 159

Figure 6.3. (a) Geometry for a nanometric electron accelerator, comprised of a glass optical fiber, which is tapered and metal coated at one end. (b) Depiction of the electric field distribution near the tip of such a device. 161

Figure 6.4. (top) Finite-difference time-domain simulation of a TEM_{00} and TEM_{10} optical pulses coupling to surfaces plasmons. In the case of the TEM_{10} mode, an amplitude modulated wave results, indicating that a spatially engineered light-wave can be implemented to switch the plasmon on a time-scale below its lifetime. (bottom) Depiction of a Bessel-Gauss mode, which, when focused, provides a longitudinal electric field that can be used to accelerate or steer electrons. 163

Figure 6.5. (top) Experimental data acquired using a titanium-sapphire oscillator in the multiphoton electron emission regime ($\gamma > 1$). The autocorrelation trace indicated no broadening due to the finite lifetime of the plasmon. In the case of Keldysh emission (bottom), electron emission would be correlated with the presence of the electric field of the plasmon. Therefore, it is expected that the autocorrelation trace would be significantly broadened as the electron emission is now driven by the electric field, which persists for the entire duration of the plasmon lifetime. 165

Figure 6.6. Conceptual illustration of a time-resolved electron diffraction experiment. A laser pump pulse is used to excite a system, and a short time later, an electron pulse probes the excited system. When the delay between the two pulses is varied, a 'motion picture' of the physical process can be created from the time-varying diffraction pattern and used to determine ultrashort dynamics of the physical process in question. 167

Figure A.1. A variable field magnetic spectrometer, comprising a central fixed-arc chamber through which electrons flow and two magnetic field generation coils required to disperse the electrons according to their velocity. Electron detection is achieved using an electron multiplier that is situated in its own housing containing the necessary electrical feedthroughs. Evacuation of the chamber occurs through a vacuum port, also attached directly to the central chamber and located at the rear. To increase the resolution of the device, an aperture, of width a, is placed within chamber in the direct path of the electrons. 173

Figure A.2. Schematic diagram of the central chamber and various adjoining components. The main chamber consists of a single fixed-arc of radius of curvature r_0, which is evacuated through a port located at 70°. Electrons, generated at the source, will follow a trajectory through the chamber that depends on the current, I, in the coils and their initial velocity. Depending on the conditions of the electron source, an optical access port is also included in the design to allow for front-illumination of samples. To restrict the number of possible electron trajectories and increase the spectrometer resolution, apertures of widths of s, d, and a are placed at the source, detector, and mid-chamber, respectively. 174

Figure A.3. Electrical schematic of the circuit used to bias the electron multiplier. Fast 1N4148 diodes are used to protect subsequent amplifier/processing electronics from high-voltage transients. 177

Figure A.4. Distribution of the $B_z(x, y)$ component of the magnetic field generated by the coils at a current of $I = 1.0$ A. The dotted line indicates the outline of the central arc in which electrons flow. 179

Figure A.5. Sample set of electron trajectories that are emitted from the source and successfully reach the detector for $I = 1.0$ A, $s = 2$ mm, $d = 1$ mm, and $a = 1$ mm, $\alpha = 60°$, and $K_0 = 50$ keV. Here, 0° represents the normal or positive y-direction. 179

Figure A.6. (a) Transmitted electron energy distribution for $I = 1.0$ A and $s = 2$ mm. Three impulse responses are shown for $a = d = 1.0$ mm, $a = d = 1.5$ mm, and $a = d = 2.0$ mm. For the 1.0 mm case, the central energy $E = 0.76$ keV and FWHM $\Delta E = 63$ eV, resulting in a resolution $\Delta E/E = 8.3$ %. For the increased values of $a = d = 1.5$ mm, and $a = d = 2.0$ mm, the uncertainty in energy measurement increases to 9.8% and 11.3%, respectively. (b) Calculated variation of the central value of energy, E, as a function of the current I (circles) for $a = d = 1.0$ mm and $s = 2$ mm. The least-squares fit (line) indicates a quadratic relationship. (c) Calculated variation of the FWHM, ΔE, as a function of the current I (circles) for $a = d = 1.0$ mm and $s = 2$ mm. The least-squares fit (line) also indicates a quadratic relationship. (d) Overall resolution of the spectrometer for $a = d = 1.0$ mm and $s = 2$ mm. 181

Figure A.7. Electron multiplier gain as a function of the externally applied magnetic field. 183

Figure B.1. Flow chart of the simulation program. 187

List of Acronyms

a. u.	Arbitrary Units
SP	surface plasmon
CEP	carrier-envelope phase
FDTD	finite-difference time-domain
PML	perfectly match layer
FWHM	full-width at half-maximum
EM	electromagnetic

Chapter 1.0

Introduction

Recent developments in optical theory and design have allowed researchers to use the laser as a tool for scientific exploration in fundamental and applied sciences as well as engineering. Within the broad field of laser science, much research has been devoted to the generation of ultrashort optical pulses [1] with the hope of investigating ultrafast events occurring on timescales of a few femtoseconds. This would lead to significant advancement of the knowledge of hidden fundamental physics within, for example, light-matter interaction, solid-state structure and dynamics, and plasma science. Using such optical pulses, ultrashort events can be studied through time-resolved experimentation: a process where single 'frames' are accumulated, pulse by pulse, to produce a 'movie' of the ultrafast process. In this technique, the temporal resolution is typically limited by the duration of the optical pulse, which at present is about 5 fs.

However, the objects or mechanisms that often mediate an ultrafast dynamical event can be confined to a spatial extent on the order of a few nanometers or less (e.g. an electron's orbit around a proton in a hydrogen nucleus). Therefore, to obtain spatial information on this length scale, the probe that is implemented must have the capability to resolve such small features. Herein lies the dilemma of the diffraction barrier postulated by E. Abbe in 1873 [2], the smallest resolvable feature that can be optically probed is approximately half the wavelength of the incident electromagnetic wave. In the case of ultrashort laser pulses produced by conventional titanium-sapphire laser systems, the wavelength of the radiation is 800 nm, and therefore, the spatial resolution would be only 400 nm.

A solution can be afforded by using an alternative technique to optical probing that is not limited by the diffraction barrier. Near-field imaging techniques can be

implemented, however, suffer from low signal levels due to the evanescent nature of the radiation and are limited to spatial resolutions of ~20 nm [3]. Rather than using photons for imaging, electrons can be employed. Electron-based imaging has the principal advantage of extremely high spatial resolution, as the wavelength of a high-energy electron can be much shorter than its photonic counterpart. Electron beam systems routinely operate with electron energies of up to 30 keV, generated via electrical biasing grids, translating to a wavelength of a mere 0.2 nm. However, the serious disadvantage of such a system is its inherent inability to convey any temporal information as it relies on a continuous beam of electrons.

1.1 Methods for Generating Electron Pulses

A simple solution to increase the temporal resolution of an electron-beam-based imaging system is to generate ultrashort energetic electron pulses, thus fulfilling both the spatial and temporal resolution requirements. While the concept is straightforward, producing electron pulses on ultrafast timescales is not trivial. Direct modulation of an electron source is limited by the switching times of high-voltage biasing electronics (sub-nanosecond) and alternative configurations must be considered. A noteworthy radio-frequency (RF) waveguide geometry has been proposed and could potentially generate 100 fs electron pulses [4] under ideal conditions, however, the design is complicated and remains to be demonstrated experimentally. In contrast to electrically based methods, the currently favored technique for ultrashort electron pulse generation relies on optical excitation and is shown in Figure 1.1. A photocathode is irradiated by a femtosecond optical pulse and emits an electron pulse, which is accelerated to the desired

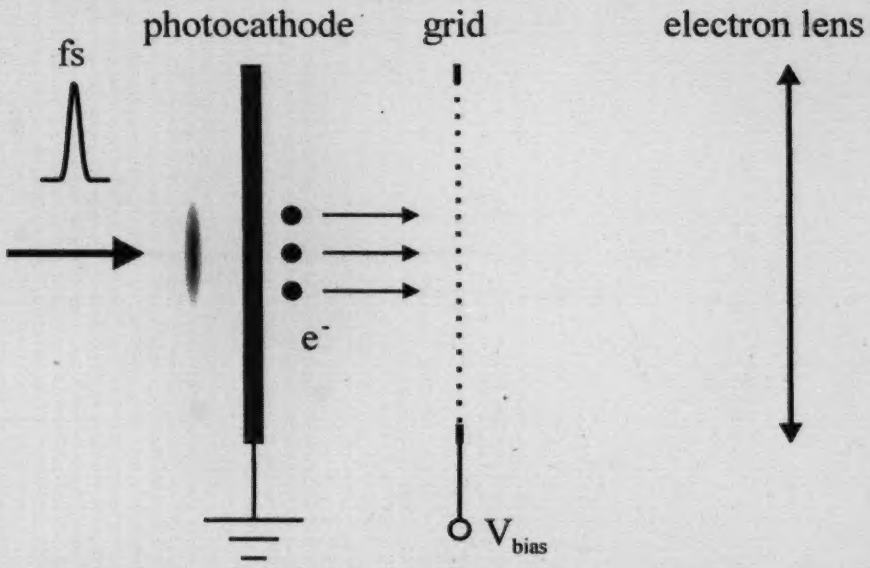

Figure 1.1. Illustration of the acceleration grid method used to generate energetic electron pulses for conventional time-resolved electron diffraction systems.

energy with an electrostatic grid biased at a large potential. Several groups [5-8] have successfully employed this method to achieved sub-picosecond electron pulse durations with energies near 30 keV. In close proximity to the surface, the duration of the electron pulse is limited only by the optical interaction, and thus, the potential for creating <100 fs electron pulses exists. However, the large and complex designs required for this form electron acceleration are fundamentally dominated by space-charge effects, which increase the electron pulse duration to a minimum of 200 fs [8-10]. The high-voltage power supplies used to bias the electron optics also pose limitations on the maximum attainable energies. For these reasons, further investigation is required to access the sub-100 fs timescale. In particular, it would be desirable to find an alternative method of all-

optical acceleration that would have the advantages of low cost, ultra-compact experimental arrangement, and short pulse duration.

As the duration of electron pulses is reduced, a related issue arises: the characterization of the ultrashort electron pulses. Electronic streaking techniques are often used to achieve electron pulse measurement and also rely on fast high-voltage transients to deflect the electron pulses. While electronic streaking has been successful for characterizing electron pulses having durations on the order of hundreds of femtoseconds [6], large beam traversal paths coupled with space-charge effects also impose limitations on the maximum attainable temporal resolution. Realizing the limitations of electrical switching, consideration is again given to optical methods. Electro-optic measurements of the electric field due to the high-charge relativistic electron bunches (~1 nC) have been demonstrated [11] with sub-100 fs temporal resolutions. However, the low sensitivity of such techniques prevents their implementation in low-charge electron pulse experiments. Other theoretical investigations propose to use optical pulses to directly disperse segments of an electron pulse [12,13]. While they show promise for breaking the 100 fs barrier, such schemes would require extremely large intensities in the >30 TW/cm^2 range to achieve the necessary electric fields (>10^8 V/cm) required for deflection of keV energy electrons.

1.2 Surface Plasmons[1]

Advances in material science and nano-engineering have rekindled early interest [14-16] in collective waves known as surface plasmons (SP), which are localized charge density

[1] This section is intended to provide a brief overview of research involving surface plasmons and is by no means exhaustive.

oscillations of the conduction band electrons of a metal. Since light can be resonantly coupled to SP's, they can be employed in a variety of technological and scientific applications spanning many different research fields.

As the SP resonance can be quite narrow (1.1°), the coupling is extremely sensitive to the surrounding material and the geometry used to couple the plasmons. Hence, SP resonance is an extremely useful tool for sensing. For instance, a large amount research has been devoted to implementing SP resonance for biological applications. A particular technique employs the nonspecific nature of SP resonance to detect molecular-pair binding at the metallic surface upon which the SP wave is launched [17]. One member of the molecular pair is localized or attached to the metal surface and forms the sensor for the other conjugate molecule. If the binding molecule is present, the local refractive index will change and a 'real-time' probing optical beam will no longer be able to couple to the SP wave. The reflected portion of such an optical probe is monitored and indicates the presence of the binding molecule, which is manifested as a shift of the plasmon resonance curve. Such devices are extremely useful for analyte detection and can be used to study antigens, DNA, enzymes, chelating agents, etc. Alternatively, the SP waves can be launched on metal nano-particles or nano-particle assembles [18] and are referred to as localized SP's. Implementing nano-particles for chemical/biological sensing is advantageous as they can be deployed inside molecular and biological systems.

Similarly, SP resonance can be used as a tool for material characterization. As shown by Pockrand [19], the complex dielectric constant and thickness of a thin film can be determined through the width, depth and location of the central dip of an SP resonance curve. Furthermore, Fontana [20] reports that surface roughness of metals can be

simultaneously measured in addition to the aforementioned material parameters. By collecting the light scattered from surface imperfections and calculating the resultant Fourier spectrum, roughness parameters can be estimated.

Aside from material and chemical characterization, a plethora of research surrounds the concept of plasmon-mediated transport. One well-recognized research article by Ebbesen et al. [21] describes extraordinary optical transmission through arrays of nanometer size holes (150 nm). This phenomenon is attributed to launching of plasmons and their subsequent propagation through the nano-metric sized metallic apertures. Such research has opened a doorway for nano-photonics and control over electromagnetic radiation.

Coupling to SP waves also provides a method for long-range energy transport within sub-wavelength systems. An excellent example is provided by the research of Maier et al. [22]. Within this work, the concept of a plasmon waveguide is presented and relies on near-field optical coupling between adjacent silver nano-rods (50 nm separation). Others yet have investigated SP propagation on thin metal strips [23] for the application of miniaturized light guides for nano-scale optics. It was observed that plasmons propagated along thin (70 nm) silver and gold surfaces for lengths of approximately 25 μm (1/e point). Similar work by the same group [24] also indicates the feasibility of implementing SP propagation for compact nanometric optical devices and probes. Here, the authors introduce confinement along another spatial direction in the investigation of gold nanowires (200 nm × 50 nm cross-section). Again, propagation lengths on the order of a few microns were measured.

Realizing the importance of confinement of electromagnetic energy during SP resonant coupling, Smolyaninov et al. [25] have studied a far-field nanoscope that is based on SP wave focusing. In a two-stage process, a far-field microscope views the local image formed by SP's on a metal film. Essentially, a dielectric droplet (shaped in the form of a parabolic lens) is used to form an image using the actual SP waves, which is then observed using a conventional microscope objective. Using a test pattern comprised of arrays of 100 nm holes in a gold surface, the authors demonstrate proof-of-principle operation of the SP-enhanced sub-wavelength imaging that could potentially break the diffraction limit by over three orders of magnitude.

Apart from pure electromagnetic interactions, examination of the behavior of photoemission from nanoclusters and nanoparticles has lead to new understanding of solid-state matter [26-30]. Electron emission is very sensitive to the local material morphology and the structure of the ionizing electric field. Therefore, knowledge of the energy and angular distribution of photoemitted electrons can be useful for determining particle distribution and work function. Furthermore, by implementing higher-order autocorrelations, information regarding SP relaxations times and plasmon dynamics can be afforded.

Despite the vast amount of research on SP's and their applications, very little work has focused on their implementation for particle acceleration/control. Such research is important for studying electromagnetic-particle interaction and may lead to significant advances in engineering and physics. Furthermore, the tools developed through such research may prove to be useful as diagnostic instruments for chemistry and biology.

1.3 Thesis Objectives

In the following chapters, an innovative scheme is described that could potentially provide both sub-100 fs electron pulse generation and measurement. The technique is based on optical excitation of surface plasmon (SP) oscillations, and represents a new field within plasmonics aside from those discussed in the previous section. A graphic illustration of SP acceleration is shown in Figure 1.2. Using an ultrashort intense laser pulse, an SP wave is generated at a metal-vacuum boundary. When electrons are subjected to the electric field of the plasmon, they are ponderomotively accelerated away from the metal surface to substantial kinetic energies (> 1 keV). Since these waves can be generated on a femtosecond timescale (<100 fs), there exists the possibility (and reality as evidenced by this work) of generating an electron pulse within an equally short time span.

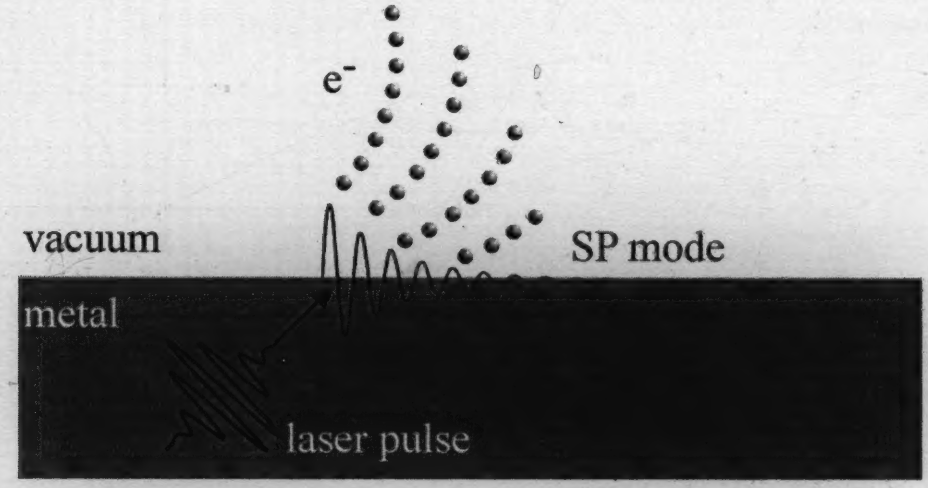

Figure 1.2. A graphical illustration of ponderomotive electron acceleration using surface plasmon waves.

In this thesis, several experiments are designed and implemented to investigate SP electron acceleration. The first broad goal is to show that this form of acceleration can be

accomplished using both low pulse energy (nJ level) and high pulse energy (mJ level) laser systems. Within this objective, the accelerated electron packets are characterized with respect to their kinetic energy spectra, angular distributions, maximum photoelectron yield, and photocurrent dependence on pump intensity. The second goal is to develop a theoretical model that accounts for the various physical mechanisms surrounding SP electron acceleration. Using this model, the energy and angular spectra of the electrons are predicted and the problem of ultrashort electron pulse duration measurement is addressed.

1.4 Applications

Before proceeding to the specific details of the generation and acceleration processes surrounding SP electron acceleration, a brief review of a few potential applications is given as motivation for this research.

a) Laser-Based Particle Acceleration

The current goal of many high-energy (> 1 GeV) particle accelerators is to uncover the structure of matter on an atomic/nuclear scale. However, many other applications also benefit from high-energy electrons. For example, synchrotrons are routinely used as broadband light sources, since a high-energy electron beam deflected through a magnetic field will yield broadband 'synchrotron radiation'. However, these sources require a vast infrastructure, and therefore, alternative methods of particle acceleration are desired. Recent work [31-33] has shown that lasers are capable of generating nearly mono-energetic electron bunches. Generation of the same quality

(emittance, pulse duration, energy) of electron bunches by using a laser source on a single tabletop would revolutionize the field, as the all-optical acceleration process would be achieved at a fraction of the cost.

b) Laser-Matter Interaction

Although a very large amount of research has been devoted to microscopic optical-matter interaction through the use of laser radiation, much more knowledge awaits that cannot be probed using laser pulses alone. Electron pulses can be employed in conjunction with laser sources to study processes such as nanoscopic lattice and carrier dynamics in semiconductors, phase transitions in solids, and nonlinear (multiphoton) ionization. In fact, recent experiments [6] using sub-picosecond electron pulses has already brought forward new physics of the melting process of aluminum due to an intense laser pulse. The potential for discovery of new physical phenomena by using even shorter electron pulses is extraordinary.

c) Coherent Optical-Particle Control

As the bandwidth-limited duration of an optical pulse generated from a titanium-sapphire system is being asymptotically approached, control over a previously inaccessible parameter has lead to a number of intriguing experiments. This parameter, known as the carrier-envelope phase (CEP), is a measure of the phase difference between peak of an optical carrier wave and its envelope. Control over this parameter can translate to direct manipulation of the outcome of a number of physical processes. For example, the generation of a single, isolated attosecond pulse [34,35] has arisen due to the precise

control the CEP of the laser pulse used to generation the high-harmonic radiation. As is demonstrated in Chapter 5, the dynamics of SP electron acceleration can be influenced by the CEP. In essence, the laser-acceleration process can be controlled through the optical waveform. On the other hand, knowledge of the laser-acceleration process itself can be used to unambiguously determine the CEP of the laser pulse. This can be applied to the measurement of the CEP of low energy pulses from a titanium-sapphire oscillator.

d) Biological Imaging: Structure-Function Relationships

Structure-function relationships in biology are one of the most intriguing, yet elusive contemporary research problems. As an example, consider a general protein structure, and the function that this particular protein achieves within its natural environment. If one can understand how the physical structure of the protein achieves it function, then one can envision any number of advances ranging from dynamical control of the protein to altering the protein to perform other functions with the end goal of creating new medicine or virus combating agents. As proteins are the building block of living matter, their structure-function relationships are paramount to understanding various organelles of cellular matter, hemoglobin, and other bio-molecular assemblies. With the availability of sub-100 fs electron pulses, the exact nature of these structure-function relationships may be uncovered, and can potentially lead to groundbreaking advances in biology, pathology, and medicine.

1.5 Thesis Organization

This thesis is comprised of six chapters and two appendices that detail the experimental and theoretical work performed. Chapter 2 introduces the underlying mechanisms involved in SP electron acceleration and provides an overview of relevant experimental and theoretical studies performed previously. Here, I review some of the preliminary experiments that focused my interest on this particular subject area and also discuss further research.

In Chapter 3, a theoretical framework is developed to investigate SP electron acceleration, which describes light-wave propagation and the influence the electric fields of the SP on electrons injected via photoemission. For this, finite-difference time-domain solution of Maxwell's equations is combined with a quasi-classical description of multiphoton emission to predict the temporal and spatial evolution of a photo-injected electron packet. Initially, this model affords a unique perspective of the ponderomotive electron acceleration process and provides a basis for comparison with experimental results.

The discussion of Chapter 4 is dedicated to the experimental aspects of SP electron acceleration. In this Chapter, the experimental arrangement is presented and the components used to study the acceleration process are described in detail. This is followed by a description of the experimental results for both laser oscillator and high-power amplifier systems. These findings are compared with the results of the theoretical model to yield further insight into the acceleration process.

Chapter 5 discusses further results obtained using the theoretical framework described in Chapter 3. The first is a novel electron beam slicing geometry. Here, it is

demonstrated that the SP wave can be used to slice an incoming electron beam to generate an ultrashort electron pulse. It is also shown that the SP wave can be used to determine the temporal envelope of a sub-100 fs electron pulse through optical-electron cross correlation. Second, the influence of the carrier-envelope phase (CEP) of a few-cycle pulse on SP electron acceleration is investigated. It is revealed that the final ponderomotive energy gain experienced by an electron can be coherently controlled through the CEP parameter of few-cycle laser pulses.

Chapter 6 summarizes the research performed in this thesis. Furthermore, future avenues of research are discussed that lead naturally from the findings presented here.

1.6 References

[1] T. Brabec and F. Krausz, "Intense few-cycle laser fields: Frontiers of nonlinear optics," *Reviews of Modern Physics* **72,** pp. 545-591 (2000).

[2] E. Abbe, "Beiträge zur Theorie des Mikroskops und der mikroskopischen Wahrnehmung," *Arch. Mikrosk. Anatomie* **9**, pp. 413 (1873).

[3] M. Ohtsu, *Near-field nano-optics: from basic principles to nano-fabrication and nano-photonics*, (Plenum Press, New York, 1999).

[4] J. Fehr, W. Reiners, L. J. Balk, E. Kubalek, D. Köther, and I. Wolff "A 100-Femtosecond Electron Beam Blanking System," *Microelectronic Engineering*, vol. **12**, pp. 221-226 (1990).

[5] A. H. Zewail, "Laser femtochemistry," *Science*, vol. **242**, pp. 1645-1653 (1988).

[6] B. J. Siwick, J. R. Dwyer, R. E. Jordan, and R. J. D. Miller, "An atomic-level view of melting using femtosecond electron diffraction," *Science*, vol. **302**, pp. 1382-1385 (2003).

[7] M. Y. Schelev, G. I. Bryukhnevich, V. I. Lozovoi, M. A. Monastyrski, A. M. Prokhorov, A. V. Smirnov, and N. S. Vorobiev, "500-fs photoelectron gun for time-resolved electron diffraction experiments," *Optical Engineering*, vol. **37**, pp. 2249-2254 (1998).

[8] J. Cao, Z. Hao, H. Park, C. Tao, D. Kau, and L. Blaszczyk, "Femtosecond electron diffraction for direct measurement of ultrafast atomic motions," *Applied Physics Letters*, vol. **83**, pp. 1044-1046 (2003).

[9] B. J. Siwick, J. R. Dwyer, R. E. Jordan, and R. J. D. Miller, "Ultrafast electron optics: Propagation dynamics of femtosecond electron packets," *Journal of Applied Physics*, vol. **92**, pp. 1643-1648 (2002).

[10] B. L. Qian and H. E. Elsayed-Ali, "Electron pulse broadening due to space charge effects in a photoelectron gun for electron diffraction and streak camera systems," *Journal of Applied Physics*, vol. **91**, pp. 462-468 (2002).

[11] I. Wilke, A. M. MacLeod, W. A. Gillespie, G. Berden, G. M. H. Knippels, and A. F. G. van der Meer, "Single-Shot Electron-Beam Bunch Length Measurements," *Physical Review Letters*, vol. **88**, pp. 124801 (2002).

[12] B. J. Siwick, A. A. Green, C. T. Hebeisen, and R. J. D. Miller, "Characterization of ultrashort electron pulses by electron-laser pulse cross correlation," *Optics Letters*, vol. **30**, pp. 1057-1059 (2005).

[13] V. I. Balykin, M. V. Subbotin, and V. S. Letokhov, "Reflection of an electron beam by femtosecond light waves," *Optics Communications*, vol. **129**, pp. 177-183 (1996).

[14] G. Mie, "Articles on the optical characteristics of turbid tubes, especially colloidal metal solutions," *Annalen Der Physik*, vol. **25**, pp. 377-445 (1908).

[15] A. Sommerfeld, "The broadening of the waves and the wireless telegraph," *Annalen Der Physik*, vol. **28**, pp. 665-736 (1909).

[16] J. Zenneck, "Breeding of even electromagnetic waves along an even conducting surface and its relation to radiotelegraphy," *Annalen Der Physik*, vol. **23**, pp. 846-866 (1907).

[17] J. Homola, S. S. Yee, and G. Gauglitz, "Surface plasmon resonance sensors: review," *Sensors and Actuators B*, vol. **54**, pp. 3-15 (1999).

[18] E. Hutter and J. H. Fendler, "Exploitation of Localized Surface Plasmon Resonance," *Advanced Materials*, vol. **16**, pp. 1685-1706 (2004).

[19] I. Pockrand, "Surface plasma oscillations at silver surfaces with thin transparent and absorbing coatings," *Surface Science*, vol. **72**, pp. 577-588 (1978).

[20] E. Fontana, "Analysis of optical surfaces by means of surface plasmon spectroscopy," *IEEE Transactions on Instrumentation and Measurement*, vol. **45**, pp. 399-405 (1996).

[21] T. W. Ebbesen, H. J. Lezec, H. F. Ghaemi, T. Thio, and P. A. Wolff "Extraordinary optical transmission through sub-wavelength hole arrays," *Nature*, vol. **391**, pp. 667-669 (1998).

[22] S. A. Maier, P. G. Kik, H. A. Atwater, S. Meltzer, E. Harel, B. E. Koel, and A. A. G. Requicha, "Local detection of electromagnetic energy transport below the diffraction limit in metal nanoparticle plasmon waveguides," *Nature Materials*, vol. **2**, pp. 229-232 (2003).

[23] B. Lamprecht, J. R. Krenn, G. Schider, H. Ditlbacher, M. Salerno, N. Felidj, A. Leitner, F. R. Aussenegg, and J. C. Weeber, "Surface plasmon propagation in microscale metal stripes," *Applied Physics Letters*, vol. **79**, pp. 51-53 (2001).

[24] J. R. Krenn, B. Lamprecht, H. Ditlbacher, G. Schider, M. Salerno, A. Leitner and F. R. Aussenegg, "Non-diffraction-limited light transport by gold nanowires," *Europhysics Letters*, vol. **60**, pp. 663-669 (2002).

[25] I. I. Smolyaninov, J. Elliott, A. V. Zayats, and C. C. Davis, "Far-Field Optical Microscopy with a Nanometer-Scale Resolution Based on the In-Plane Image Magnification by Surface Plasmon Polaritons," *Physical Review Letters*, vol. **94**, pp. 057401 (2005).

[26] F. Megi, M. Belkacem, M. A. Bouchene, E. Suraud, and G. Zwicknagel, "On the importance of damping phenomena in clusters irradiated by intense laser fields," *Journal of Physics B*, vol. **36**, pp. 273-282 (2003).

[27] L. Köller, M. Schumacher, J. Köhn, S. Teuber, J. Tiggesbäumker, and K. H. Meiwes-Broer, "Plasmon-Enhanced Multi-Ionization of Small Metal Clusters in Strong Femtosecond Laser Fields," *Physical Review Letters*, vol. **82**, pp. 3783-3786 (1999).

[28] J. Lehmann, M. Merschdorf, W. Pfeiffer, A. Thon, S. Voll, and G. Gerber, "Surface Plasmon Dynamics in Silver Nanoparticles Studied by Femtosecond

Time-Resolved Photoemission," *Physical Review Letters*, vol. **85**, pp. 2921-2924 (2000).

[29] J.-H. Klein-Wiele, P. Simon, and H.-G. Rubahn, "Size-Dependent Plasmon Lifetimes and Electron-Phonon Coupling Time Constants for Surface Bound Na Clusters," *Physical Review Letters*, vol. **80**, pp. 45-48 (1998).

[30] B. Lamprecht, A. Leitner, and F. R. Aussenegg, "Femtosecond decay-time measurement of electron-plasma oscillation in nanolithographically designed silver particles," *Applied Physics B*, vol. **64**, pp. 269-272 (1997).

[31] S. P. D. Mangles, C. D. Murphy, Z. Najmudin, A. G. R. Thomas, J. L. Collier, A. E. Dangor, E. J. Divall, P. S. Foster, J. G. Gallacher, C. J. Hooker, D. A. Jaroszynski, A. J. Langley, W. B. Mori, P. A. Norreys, F. S. Tsung, R. Viskup, B. R. Walton, and K. Krushelnick, "Monoenergetic beams of relativistic electrons from intense laser-plasma interactions," *Nature*, vol. **431**, pp. 535-538 (2004).

[32] C. G. R. Geddes, C. S. Toth, J. Van Tilborg, E. Esarey, C. B. Schroeder, D. Bruhwiler, C. Nieter, J. Cary, and W. P. Leemans, "High-quality electron beams from a laser wakefield accelerator using plasma-channel guiding," *Nature*, vol. **431**, pp. 538-541 (2004).

[33] J. Faure, Y. Glinec, A. Pukhov, S. Kiselev, S. Gordienko, E. Lefebvre, J.-P. Rousseau, F. Burgy, and V. Malka, "A laser-plasma accelerator producing monoenergetic electron beams," *Nature*, vol. **431**, pp. 541-544 (2004).

[34] P. Agostini and L. F. DiMauro, "The physics of attosecond light pulses," *Reports on Progress in Physics*, vol. **67**, pp. 813-855 (2004).

[35] A. Baltuška, Th. Udem, M. Uiberacker, M. Hentschel, E. Goulielmakis, Ch. Gohle, R. Holzwarth, V. S. Yakovlev, A. Scrinzi, T. W. Hänsch, and F. Krausz, "Attosecond control of electronic processes by intense light fields," *Nature*, vol. **421**, pp. 611-615 (2003).

Chapter 2.0

Background

To understand surface plasmon electron acceleration, one must understand several fundamental underlying physical concepts including laser-plasmon coupling, electron emission, and ponderomotive acceleration. The purpose of this chapter is to outline these processes in detail. First, the Drude model, which describes the response of free electrons in a metal to an externally applied electric field, is discussed. This naturally leads to the concept of a surface plasmon (SP) wave, which is explained in the following section. A physical description of the electric and magnetic fields associated with this wave as well as its dispersion relation is derived from Maxwell's equation. After the discussion of coupling to SP waves, electron emission mechanisms and the ponderomotive force are described. Finally, a literature survey of relevant work is presented.

2.1 Drude Theory of Metals

To derive the frequency dependent dielectric function of a metal film, consider a sea of free electrons subjected to an external driving electric field [1,2]. The one-dimensional equation of motion of a single electron can be written in terms of a displacement $x(t)$:

$$\frac{d^2x}{dt^2} + \nu_d \frac{dx}{dt} = -\frac{qE(t)}{m_e}, \tag{2.1}$$

where $E(t)$ is the externally applied electric field, ν_d is the damping rate (due to electron collision), and q and m_e are the charge and mass of the electron, respectively. Solution of this equation can be found through the usual ansatz of harmonic time dependence, where $x = x_0 e^{i\omega t}$ and $E = E_0 e^{i\omega t}$ are substituted into equation 2.1 to yield the following relationship between the displacement and electric field amplitude:

$$x_0 = -\frac{qE_0}{m_e}\frac{1}{i\omega\nu_d - \omega^2}. \tag{2.2}$$

The polarization response of the material to this electric field is the sum over all electric dipoles per unit volume, which in the case of a metal can be written as $P(t) = -n_e q x(t)$, where n_e is the electron number density. This equation, in conjunction with equation 2.2, can be used to derive the electric susceptibility:

$$\chi = \frac{P(t)}{E(t)} = \frac{n_e q^2}{m_e}\left(\frac{1}{i\omega\nu_d - \omega^2}\right). \tag{2.3}$$

Therefore, the dielectric function $\varepsilon = \varepsilon_0 + \chi$ can be written as

$$\varepsilon = \varepsilon_0\left(1 - \frac{\omega_p^2}{\omega^2 - i\omega\nu_d}\right), \tag{2.4}$$

where $\omega_p = \sqrt{n_e q^2/\varepsilon_0 m_e}$ denotes the plasma frequency and ε_0 is the permittivity of free space. The real, ε_r, and imaginary, ε_i, components of the dielectric function are plotted in Figure 2.1 along with the refractive index, n, and absorption coefficient, k. It is observed that the medium is highly absorbing below the plasma frequency, which results in an almost perfect reflection of an incident electromagnetic wave. Above the plasma frequency, the material simply behaves as a dielectric, with an index close to unity and a vanishing absorption coefficient. As shown in the next section, it is the negative dielectric function (i.e. directly below ω_p) that is necessary for launching of SP waves.

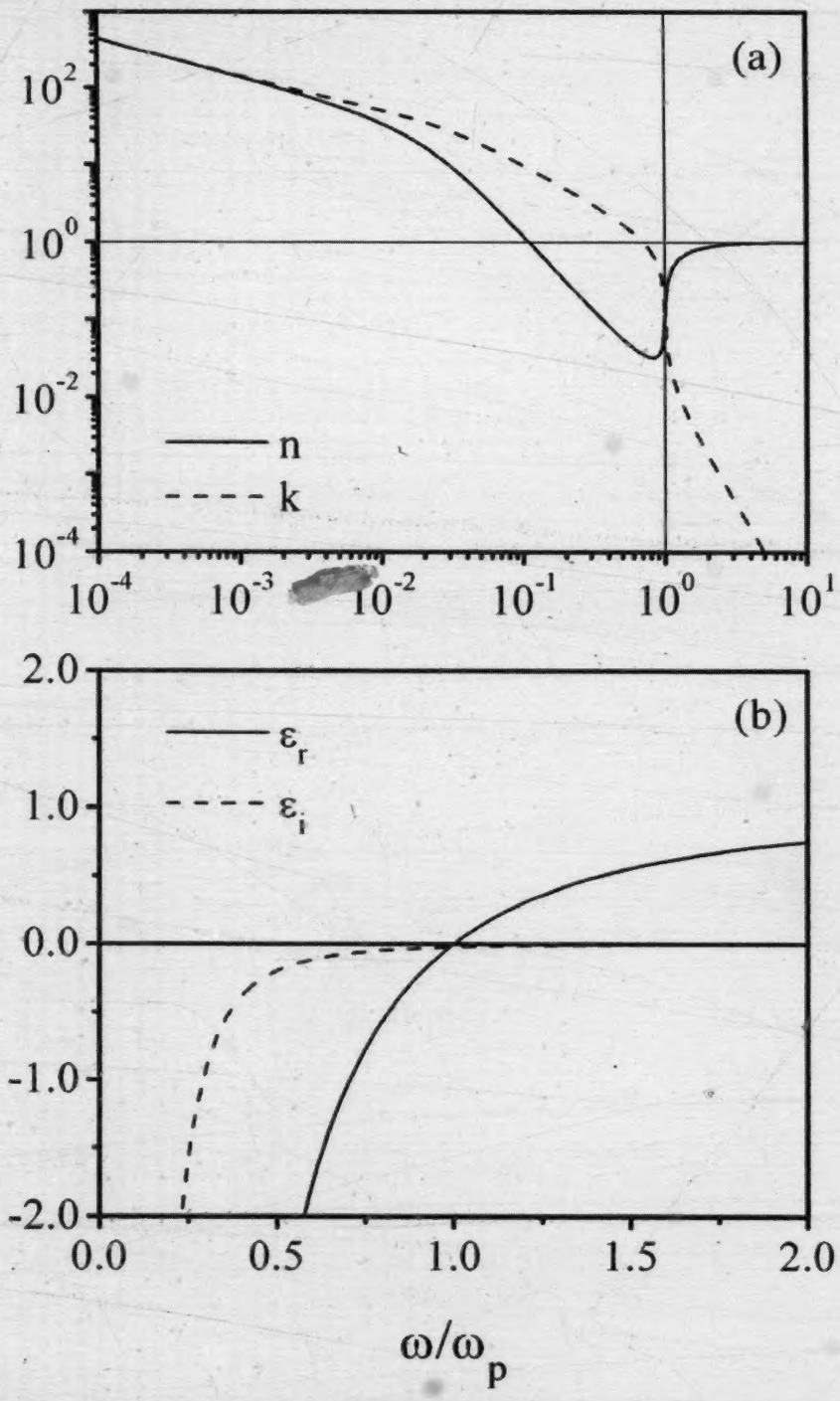

Figure 2.1. Optical constants (a) and dielectric function (b) as calculated from the Drude model.

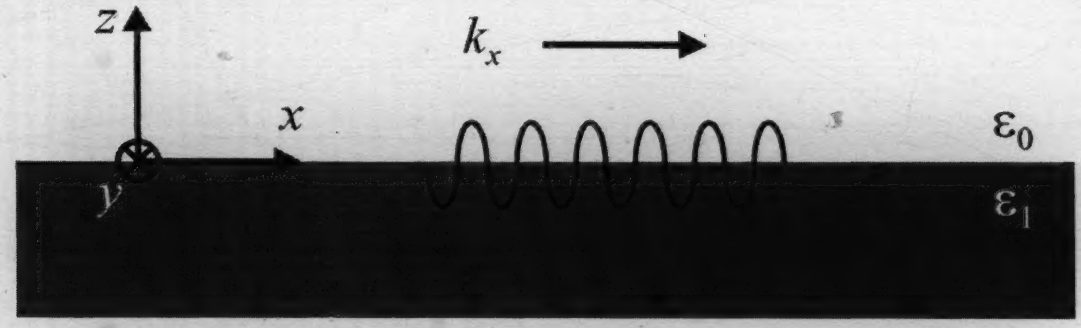

Figure 2.2. Two dielectric half-spaces which are divided at $z = 0$. At the interface, an SP oscillation, propagating along the positive x-direction, can exist.

2.2 Surface Plasmon Waves

A plasmon wave is a coherent oscillation of the conduction band electrons of a metal. Clearly, a *surface* plasmon (SP) is a charge oscillation that is confined to a metal surface, more specifically, a dielectric-metal interface. The situation is depicted in Figure 2.2, which illustrates two adjacent half-spaces of arbitrary dielectric constants. Above the plane $z = 0$, space is filled with vacuum having a dielectric constant ε_0, while for $z < 0$ the material is assigned a permittivity ε_1. Intuitively, the surface charge oscillation exists at the boundary between the two materials and must obey Maxwell's equations. To derive an expression for the electric field distribution of such an oscillation as well as its dispersion relation, the treatment presented in [1] is followed. A solution of Maxwell's equation is sought after, in which the electric field is confined to the interface at $z = 0$ and propagates in the positive x-direction. Therefore, it is assumed that the electric, $\bar{E}$, and magnetic, $\bar{H}$, fields of the SP have a form:

$$\bar{E} = (E_1, E_2, E_3)\, e^{i(k_x x - \omega t)} e^{-\alpha_0 z} \qquad z > 0 \qquad (2.5)$$

$$\bar{E} = (E_4, E_5, E_6)\, e^{i(k_x x - \omega t)} e^{\alpha_1 z} \qquad z < 0 \qquad (2.6)$$

and

$$\bar{H} = (H_1, H_2, H_3)\, e^{i(k_x x - \omega t)} e^{-\alpha_0 z} \qquad z > 0 \qquad (2.7)$$

$$\bar{H} = (H_4, H_5, H_6)\, e^{i(k_x x - \omega t)} e^{\alpha_1 z} \qquad z < 0, \qquad (2.8)$$

respectively. Here, α_0 and α_1 are positive to ensure that the wave is confined to the interface at $z = 0$ and that the electric field decays exponentially away from the interface. Boundary conditions require that the tangential field components are continuous across the interface, and thus it can be observed immediately that $E_1 = E_4$, $E_2 = E_5$, $H_1 = H_4$, and $H_2 = H_5$. Since the material is non-magnetic, i.e. $\mu = \mu_0$, the continuity on the normal component of $\bar{B}$ indicates that $H_3 = H_6$. Furthermore, $\nabla \cdot \bar{B} = 0$ implies that $H_1 = H_3 = 0$ and thus, the fields can be written as:

$$\bar{E} = (E_1, E_2, E_3)\, e^{i(k_x x - \omega t)} e^{-\alpha_0 z} \qquad z > 0 \qquad (2.9)$$

$$\bar{E} = (E_1, E_2, E_6)\, e^{i(k_x x - \omega t)} e^{\alpha_1 z} \qquad z < 0 \qquad (2.10)$$

and

$$\bar{H} = (0, H, 0)\, e^{i(k_x x - \omega t)} e^{-\alpha_0 z} \qquad z > 0 \qquad (2.11)$$

$$\bar{H} = (0, H, 0)\, e^{i(k_x x - \omega t)} e^{\alpha_1 z} \qquad z < 0, \qquad (2.12)$$

where $H = H_2 = H_5$. These equations are then substituted into Ampere's law relating the time-varying electric and magnetic fields, $\nabla \times \bar{H} = \partial \bar{D} / \partial t$, which is given in component form:

$$\left(-\frac{\partial H_y}{\partial z}, 0, \frac{\partial H_y}{\partial x} \right) = -\varepsilon_1 i\omega (E_x, E_y, E_z) \qquad (2.13)$$

It is observed that the E_y component must be zero (since $\bar{H}$ has only a *y*-component), which dictates that $E_2 = 0$ in equations 2.9 and 2.10. This is an interesting and useful point: the SP mode has a transverse magnetic (TM) polarization. A transverse electric (TE) polarization SP mode cannot satisfy Maxwell's equations and the boundary conditions simultaneously. Substitution of the fields in equations 2.9-2.12 into equation 2.13 yields

$$(-\alpha_0 H, 0, -k_x H) = \varepsilon_0 i\omega (E_1, 0, E_2) \qquad z > 0 \tag{2.14}$$

and

$$(\alpha_1 H, 0, -k_x H) = \varepsilon_1 i\omega (E_1, 0, E_6) \qquad z < 0 . \tag{2.15}$$

Examination of the *x*-components of these equations reveals $-\alpha_0 H = \varepsilon_0 i\omega E_1$ and $\alpha_1 H = \varepsilon_1 i\omega E_1$, and when divided, gives:

$$\frac{\alpha_0}{\alpha_1} = -\frac{\varepsilon_0}{\varepsilon_1} . \tag{2.16}$$

The above result is particularly enlightening: since α_0 and α_1 are assumed to be positive (to satisfy the wave confinement), a positive dielectric constant ε_0 (vacuum) implies that the dielectric constant of the second medium, ε_1, must be negative. As already noted in the previous section, metals have a negative dielectric function for a range of frequencies below the plasma frequency, ω_p.

The dispersion relation of an SP mode can be obtained through the wave equation:

$$\nabla^2 \bar{E} = \varepsilon \mu_0 \frac{\partial^2 \bar{E}}{\partial t^2} . \tag{2.17}$$

Direct substitution of equations 2.9 to 2.12 into equation 2.17 results in the following expressions:

$$-k_x^2 + \alpha_0^2 = -\varepsilon_0 \mu_0 \omega^2 \qquad z > 0 \tag{2.18}$$

and

$$-k_x^2 + \alpha_1^2 = -\varepsilon_1 \mu_0 \omega^2 \qquad z < 0\,. \tag{2.19}$$

These equations are then rearranged and their quotient gives

$$\frac{\alpha_0^2}{\alpha_1^2} = \frac{k_x^2 - \varepsilon_0 \mu_0 \omega^2}{k_x^2 - \varepsilon_1 \mu_0 \omega^2}\,. \tag{2.20}$$

Finally, equation 2.16 is inserted into equation 2.20 to result in an explicit expression relating the frequency of an SP mode to its wavevector:

$$\omega = c k_x \sqrt{\frac{(\varepsilon_0 + \varepsilon_1)}{\varepsilon_0 \varepsilon_1}}\,. \tag{2.21}$$

Figure 2.3 illustrates a plot of the dispersion curves of a photon and an SP wave. Here it is assumed that the dielectric function of the metal (ε_1) is given by the Drude model (equation 2.4) and $\nu_d = 0$. In the large wavevector limit (short wavelength), it is observed that the frequency approaches an asymptotic value of $\omega_p / \sqrt{\varepsilon_0 + 1}$, which is often referred to as the surface plasmon frequency. Also of significant interest is the low-frequency behavior of the SP dispersion relation. As the wavevector approaches zero, the dispersion curve asymptotically approaches the dispersion function of a photon, however, the two curves never actually cross. Therefore, direct coupling between electromagnetic waves and SP oscillations cannot occur, as there is no matching of the wavevectors at a given frequency.

2.3 Electromagnetic Wave Coupling to Surface Plasmons

In order to launch an SP wave using laser radiation, the dispersion curves of the photon and SP must be made to cross to conserve both energy and momentum in the interaction. One such method is known as prism coupling or the Kretschmann configuration. A schematic diagram of this technique is shown in Figure 2.4. A thin metal film of thickness d, which will support the SP wave, is placed in direct contact with the hypotenuse face of a right angle prism. Light to be coupled to the SP wave enters the prism and impinges on the backside of the metal film. This technique make use of the fact that the slope of the photon dispersion function is lowered by the refractive index of the prism and then intersects the SP dispersion curve (see Figure 2.3). The frequency of the oscillation, as it is converted to an SP mode, remains constant. However, the component of the propagating wavevector that is projected onto the interface is required to match the wavevector of the SP mode and occurs at an angle given by:

$$k_x = k_{photon} \sin(\theta_{SP}) = \frac{\omega}{c} n_{prism} \sin(\theta_{SP}), \tag{2.22}$$

where n_{prism} is the refractive index of the prism and θ_{SP} is the SP resonance angle. In this manner, light can be transformed from a free-space electromagnetic wave to a SP mode. As described in the previous section, the polarization of the incident radiation must be TM in order to satisfy the boundary conditions of the SP wave. Notably, the electric field of the plasmon wave is often described in terms of the coupling laser field as: $\bar{E}_{SP} = \eta \bar{E}_{laser} e^{i(k_x x - \omega t)} e^{-\alpha_0 z}$. Here, the empirical enhancement factor, η, represents the fact the energy of the electromagnetic wave is localized to the metal-vacuum interface, which

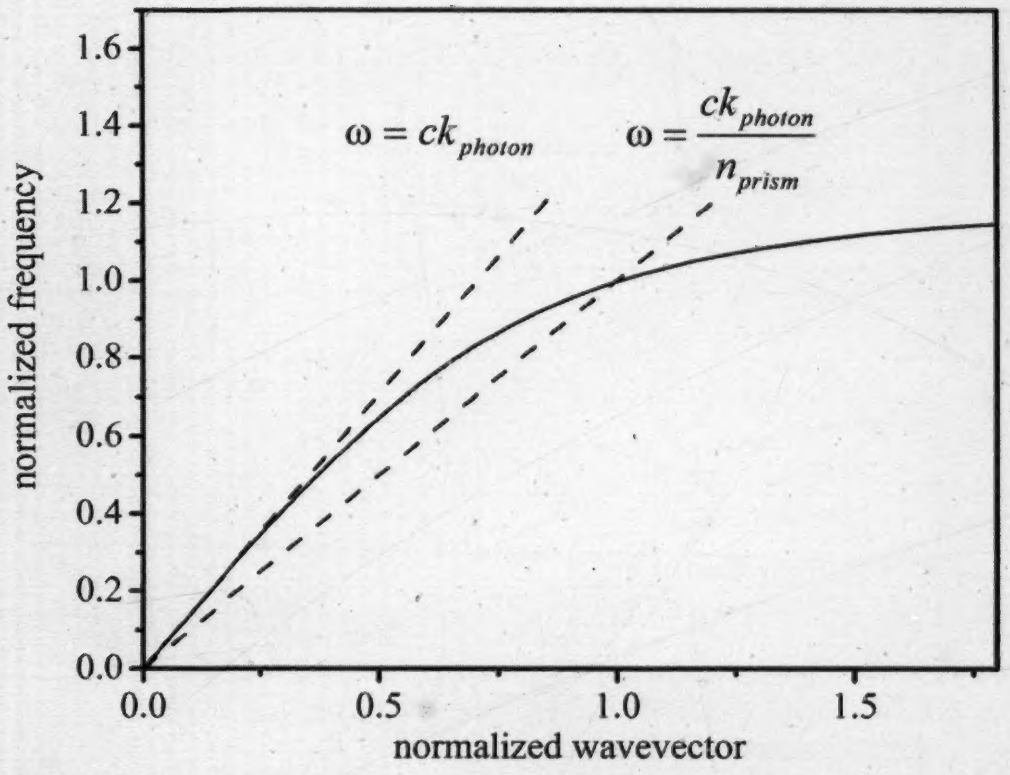

Figure 2.3. (solid line) Dispersion curve of a surface plasmon oscillation. Photon dispersion curves are also shown (dashed lines) for the cases of free-space mode and a wave traveling in a dielectric medium. The axes are normalized to the SP frequency and wavevector.

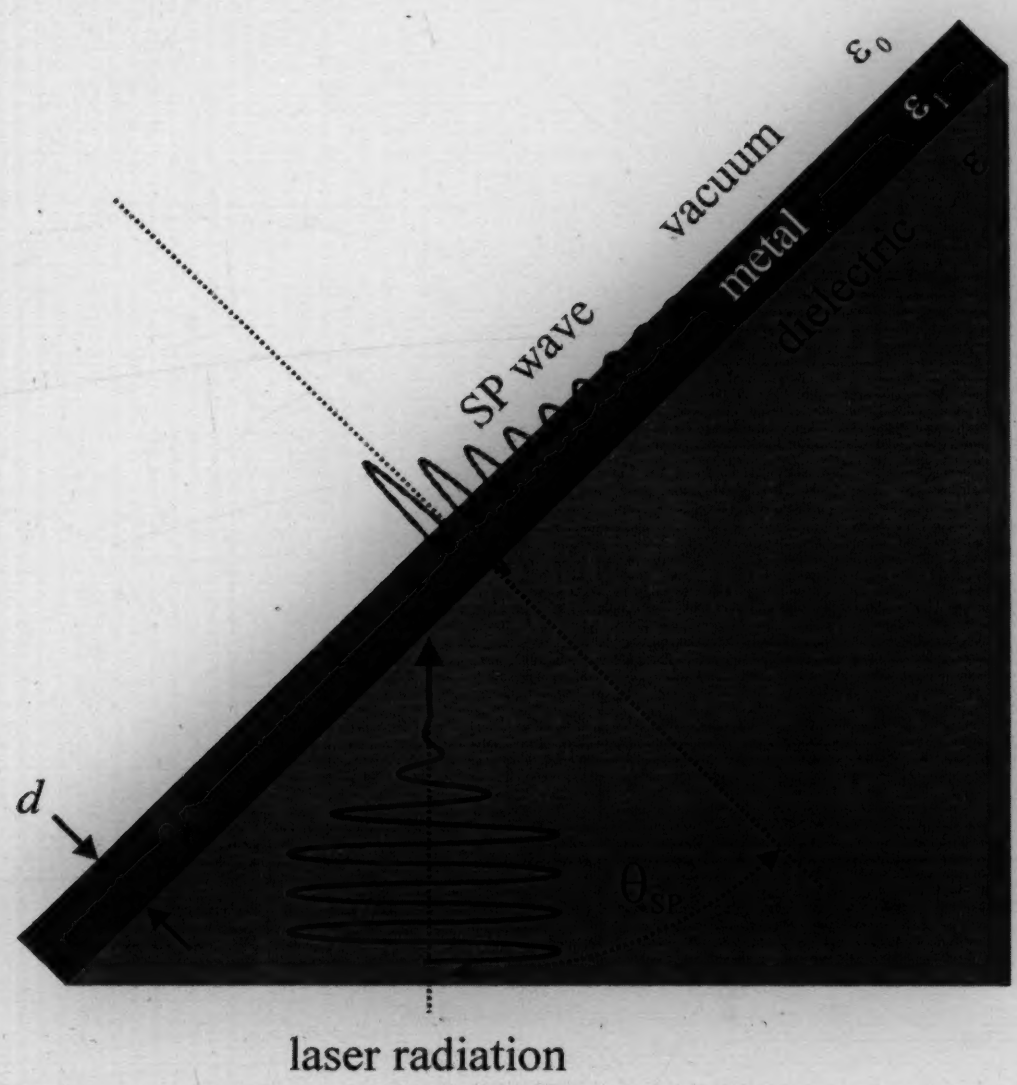

Figure 2.4. Prism geometry for coupling free-space electromagnetic waves to surface plasmon oscillations, also known as the Kretschmann configuration.

results in an enhanced electric field. The evanescent penetration depth into vacuum, α_0, can be determined by inserting equation 2.22 into equation 2.18 to arrive at the expression

$$\alpha_0 = \frac{\omega}{c}\sqrt{n_{prism}^2 \sin^2(\theta_{SP}) - 1}\,. \tag{2.23}$$

For laser radiation having a central wavelength of $\lambda_0 = 800$ nm, a prism with n_{prism}=1.59, and $\theta_{SP} = 45°$, a penetration depth of $\alpha_0^{-1} = 243$ nm is calculated. As described later, it is evanescent decay in combination with the large electric field enhancement that provides a large spatial field gradient for ponderomotive acceleration of electrons.

The thickness of the metal film will have a profound effect on the coupling efficiency. To understand the effect of varying the film thickness and to find an optimum d for SP coupling, Fresnel's coefficients for the multilayer structure can be derived. As shown in the top of Figure 2.5, multiple reflections and transmissions must be considered, which is analogous to the physical description of a Fabry-Perot interferometer. Calculation of the reflectivity, r_{210}, of the multilayer structure requires each of the coherent multiple beams must be summed. Consideration must also be given to the phase incurred as a result of propagation through medium 1 (metal) and the various transmission and reflection coefficients associated with the 21 and 10 interfaces. A plane wave dependence for the electric field is assumed and results in an expression of the form

$$r_{210} = \frac{E_r}{E_0} = E_1 + E_2 + E_3 + E_4 \ldots\,, \tag{2.24}$$

where E_n are the multiple reflections from the interfaces and are given by

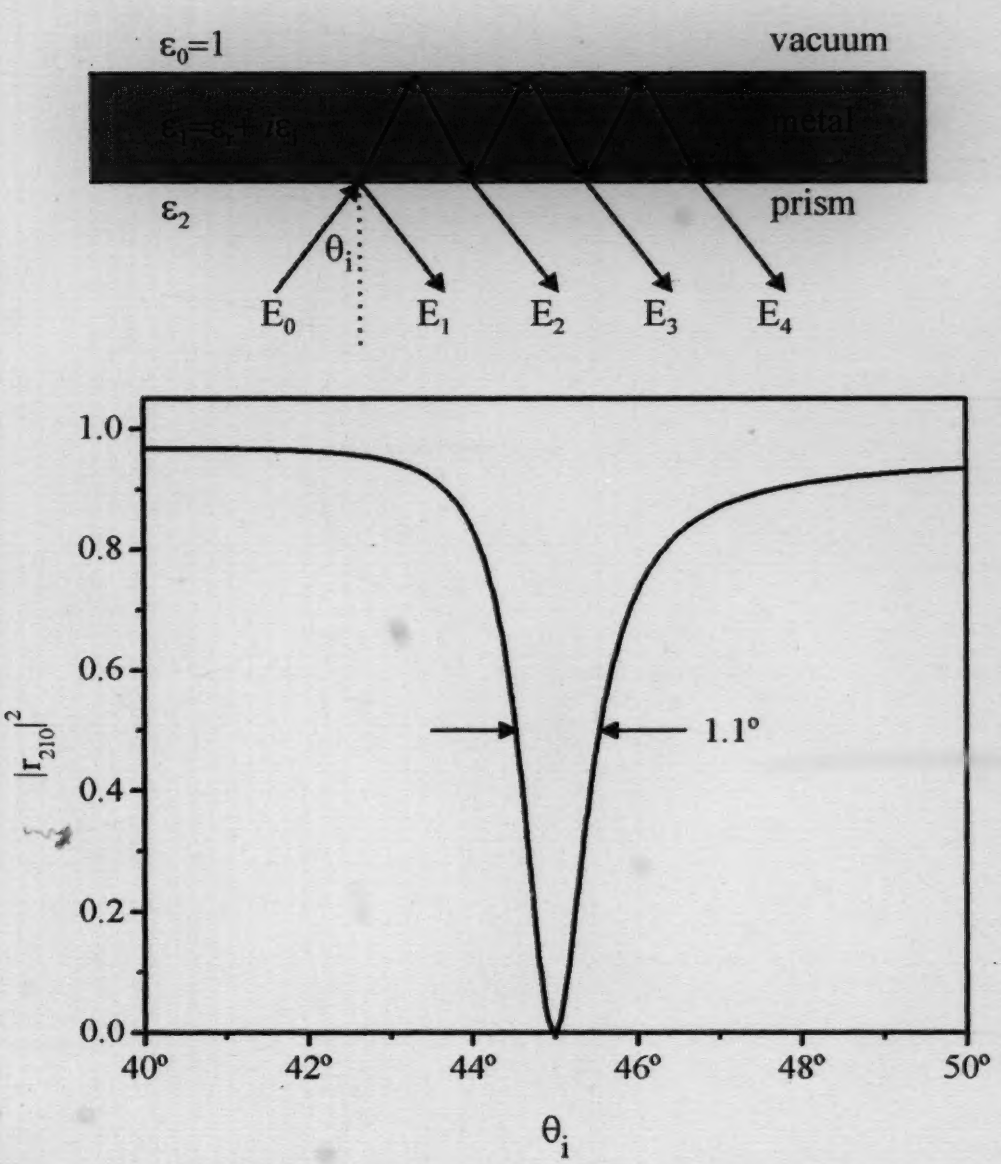

Figure 2.5. (top) Three-layer system comprised of vacuum-metal-dielectric media. Portions of an incident beam experience multiple reflections within the metallic film, which must be summed to calculate the effective Fresnel coefficient of reflection, r_{210}, for the entire structure. (bottom) Reflection coefficient of the multi-layer structure as a function of angle, clearly illustrating resonance absorption of the incident beam and coupling to SP waves.

$$
\begin{aligned}
E_1 &= r_{21} \\
E_2 &= t_{21}t'_{21}r_{10}e^{i\delta} \\
E_3 &= t_{21}t'_{21}r'_{21}(r_{10})^2 e^{i2\delta} \\
E_4 &= t_{21}t'_{21}(r'_{21})^2(r_{10})^3 e^{i3\delta} \\
&\ldots
\end{aligned}
\qquad (2.25)
$$

Here, t_{10} and t_{21} are the Fresnel transmission coefficients and r_{10} and r_{21} represent the reflection coefficients, for the respective 10 and 21 boundaries. The primed variables indicate that the traversal of the ray across the particular boundary is opposite to the direction of propagation of the incident ray. The following relations exist for the primed and unprimed coefficients: $r_{21} = -r'_{21}$ and $t_{21}t'_{21} = 1-(r_{21})^2$. Additional phase shifts present in equation 2.25 accumulate due to propagation through the metal film and are represented by the $e^{i\delta}$ factors, where $\delta = 2\pi n_1 d\lambda_0^{-1}\cos(\theta_t)$, n_1 is the complex refractive index of the metal, and θ_t is the angle of the transmitted ray measured from the normal of the surface. The ℓ^{th} beam within the infinite series of equation 2.24 can be written as

$$
r_{210} = r_{21} + t_{21}t'_{21}r_{10}e^{-i\delta}\sum_{\ell=0}^{\infty}(r'_{21})^{\ell}(r_{10})^{\ell}e^{i\ell\delta}. \qquad (2.26)
$$

Such a geometric progression has a close form sum of

$$
r_{210} = r_{21} + t_{21}t'_{21}r_{10}e^{i\delta}\left(\frac{1}{1-r'_{21}r_{10}e^{i\delta}}\right), \qquad (2.27)
$$

which can be rearranged to yield

$$
r_{210} = \frac{r_{21} + r_{10}e^{i\delta}}{1 + r_{21}r_{10}e^{i\delta}}. \qquad (2.28)
$$

The quantity relevant to SP studies is the reflectance, or simply $|r_{210}|^2$, which can be developed further (after much algebra and a few approximations) to yield [3] a more intuitive Lorentzian form

$$|r_{210}|^2 = 1 - \frac{4\Gamma_i\Gamma_r}{\left(k_{photon} - \mathrm{Re}(k_x)\right)^2 + \left(\Gamma_i + \Gamma_r\right)^2}, \tag{2.29}$$

where $\Gamma_i = \mathrm{Im}(k_x)$ represents the ohmic losses of the metal film. When the component of the incident wavevector in the film plane $k_{photon} = k_0 n_{prism} \sin(\theta_i)$, where θ_i is the angle of incidence, equals the real part of the plasmon wavevector k_x, the maximum of the resonance condition is satisfied and $|r_{210}|^2$ is minimized. It is also observed that a reflectance of zero can be achieved when the material losses, Γ_i, are balanced by the geometrical losses, Γ_r, the latter of which is controlled through the film thickness according to the approximate [4] expression:

$$\Gamma_r = \frac{\omega}{c}\mathrm{Im}(r_{21})\left(\frac{2}{\varepsilon_0 - \mathrm{Re}(\varepsilon_1)}\right)\left(\frac{\mathrm{Re}(\varepsilon_1)\varepsilon_0}{\mathrm{Re}(\varepsilon_1) + \varepsilon_0}\right)^{\frac{3}{2}} e^{\frac{4\pi d}{\lambda_0}\frac{\mathrm{Re}(\varepsilon_1)}{\sqrt{-\mathrm{Re}(\varepsilon_1)-\varepsilon_0}}} \tag{2.30}$$

Thus, an effective impedance matching condition can be satisfied in which the two forms of damping are balanced. Equation 2.28 is illustrated in Figure 2.5, where the dielectric constant is given by the Drude model (equation 2.4), with $\omega_p = 5.66\times10^{15}$ Hz and $v_d = 6.3\times10^{13}$ Hz, and the optimized film thickness is 75 nm. The SP resonance behavior is clearly evidenced by the dip in the reflectivity curve, located at an angle of 45° (for n_{prism}=1.59) with a full-width at half-maximum of 1.1°. Far away from resonance, almost no energy is coupled to the SP mode as evidenced by the fact the fraction of reflected light approaches unity.

2.4 Electron Emission from Metallic Surfaces

Experiments investigating electron emission from metal surfaces date back to 1916, the period in which Millikan first characterized the photoelectric effect [5]. For the linear photoelectric effect, a single photon from a light source can eject a single electron from metal surface, provided that the energy of the photon is larger than the work function of the metal. Since the work functions of typical metals range up to a few electron volts, the wavelength of the radiation typically lies in the ultraviolet portion of the electromagnetic spectrum. It is well known, the number of photoelectrons liberated from a metal surface is linearly proportional to the intensity of the incident radiation. However, for intense optical beams, nonlinear photoemissions mechanisms can also occur, the simplest of which is multiphoton absorption. In such cases, two or more photons can be sequentially absorbed and eject a single electron. The energy of each individual photon is lower than the work function of the metal, however, the energies of multiple photons can be added to eject a single electron. The energy balance of such an interaction obeys the well-known Einstein formula for multiphoton electron emission:

$$\Phi_m = m\hbar\omega - W_f, \tag{2.31}$$

where m is the number of photons required to photo-eject a single electron, W_f is the work function of the metal, and Φ_m is the remaining energy imparted to the electron as kinetic energy. For multiphoton photoemission, the number of photoelectrons generated in the process no longer scales linearly with the intensity of the light source, but rather scales as I_{laser}^m. The origin of this dependence can be realized schematically in Figure 2.6. For linear absorption cases, the rate of photoelectron generation, or photocurrent i_{photo}, is

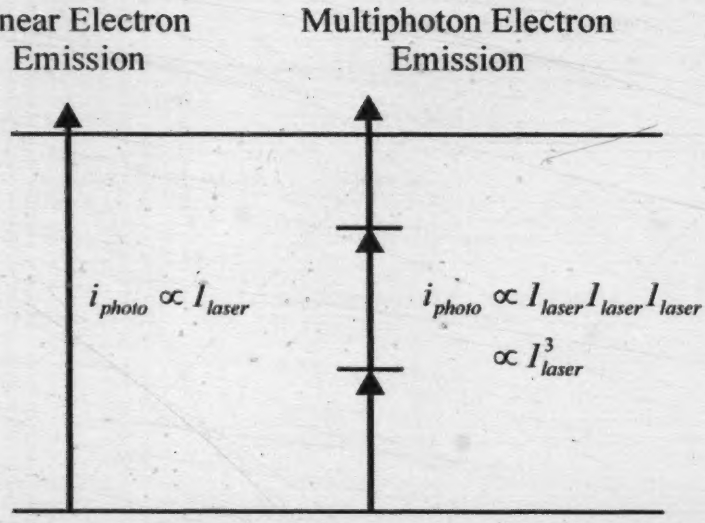

Figure 2.6. Comparison of linear (left) and multiphoton (right) electron emission from a metallic surface. For any single electronic transition, the probability of electron excitation is linearly proportional to the intensity of the exciting laser radiation. When multiple (virtual) transitions are involved, the scaling relationship is an integer power that is equal to the number of virtual transitions.

proportional to the photon density, or simply the intensity, I_{laser}, of the incident light beam. The multiphoton photoemission process can be viewed as a number of incremental linear photoemission processes that take place at various intermediate virtual states. At each stage, the probability of an electron being raised to the subsequent energy level is linearly proportional to the intensity. Since m levels are required for a bound-free transition, the number of photo-generated electrons scales as I_{laser}^m.

Early investigations by Tsang et al. in 1991 [6] have shown that the quantum efficiency of the multiphoton photoemission process can be greatly enhanced by coupling to SP waves, a distinct advantage that permits the generation of more electrons with a lower-energy laser pulse. In fact, enhancements in the quantum efficiency by more than three orders of magnitude have been observed for electron emission from silver and gold

films. The source of this enhancement is attributed to the fact the SP wave increases the photon density within the metal film, since nearly all the incident radiation can be coupled to the metal-vacuum interface. That is, rather than coupling only a few percent of the incident beam, as would be the case for normal incidence on a metal surface, SP coupling allows for nearly 100% coupling efficiency.

As the intensity of the laser radiation is increased even further, another nonlinear photoemission mechanism can occur that competes with multiphoton electron emission. This process is known as field emission, whereby an electron can undergo quantum tunneling through the potential barrier within the metal. In such cases, the electric field of the laser is large enough to reduce the height of the potential, which is given by the work function of the metal, W_f, and cause electrons to tunnel directly into vacuum. This is a well-known phenomenon that occurs within the context of photo-ionization of atoms in an intense laser field [7]. Here, the transition from the multiphoton regime to the tunnel or Keldysh regime is often characterized by the Keldysh adiabaticity parameter:

$$\gamma = \frac{\omega\sqrt{2m_e I_{pot}}}{qE_{laser}}, \tag{2.32}$$

where E_{laser} is the electric field of the photo-ejecting radiation, and I_{pot} represents the ionization potential of the atomic species. Values of $\gamma > 1$ indicate that the multiphoton process will dominate the photoemission, whereas $\gamma < 1$ indicates that tunnel emission will be favored. However, recent work [8], confirming previous theoretical studies [9], indicates that the Keldysh theory is also relevant to photoemission of electrons from metallic surfaces. In such cases, the Keldysh parameter is still useful, however, the ionization potential is replaced with the work function of the metal, W_f. Indeed, the

experimental results shown in Chapter 4 of this thesis also provide evidence for the existence of this phenomenon.

2.5 The Ponderomotive Force

Once emitted from the metal film, electrons will feel the ponderomotive force of the evanescent SP wave. This nonlinear interaction occurs in the presence of an electromagnetic field gradient, that is, an electromagnetic wave that has a spatially varying amplitude. A geometrical illustration of ponderomotive SP electron acceleration is shown in Figure 2.7 and is explained as follows. During successive oscillations of the

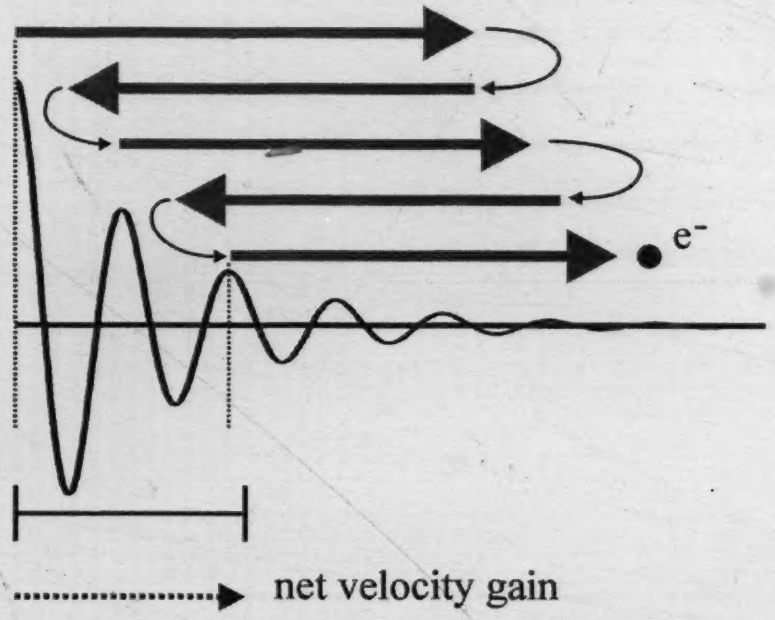

Figure 2.7. Illustration of the process of electron acceleration via the ponderomotive force. Electrons, when subjected to a spatial electric field gradient, will feel asymmetric forces in the subsequent oscillations of the wave. This results in a net gain in velocity in the direction of decreasing electric field amplitude. In the case of a surface plasmon, the direction is away from the metal film surface.

SP wave, an electron experiences asymmetric forces commensurate with the exponential decay of the SP electric field. Essentially, the electron is 'pushed' by the electric force in the direction of decreasing field amplitude by an amount that is larger than the following cycle in which it is 'pulled' back. The result is a net increase in the electron's velocity along the direction of decreasing electric field amplitude. In the time-average over many oscillations, the low-frequency ponderomotive force emerges in a direction normal to the prism surface and imparts a kinetic energy to the electron. If no such spatial gradient were present, the electron would merely experience symmetric, but opposing forces that would cancel over time.

To derive an analytic expression for the ponderomotive force, the Lorentz force equation can be implemented by considering the electric field contribution as first order, and then adding the magnetic field contribution as a perturbation [10]. Proceeding in this fashion and assuming a simple laser electric field dependence of the form $\bar{E} = \bar{E}_0(r)\cos(\omega t)$, the Lorentz force for an electron is:

$$m_e \frac{d\bar{v}_1}{dt} = -q\bar{E}_0(\bar{r}_0)\cos(\omega t). \tag{2.33}$$

This equation can be integrated to yield a velocity

$$\bar{v}_1 = -\frac{q}{m_e\omega}\bar{E}_0(\bar{r}_0)\sin(\omega t) \tag{2.34}$$

and an incremental change in position

$$\delta\bar{r} = \frac{q}{m_e\omega^2}\bar{E}_0(\bar{r}_0)\cos(\omega t). \tag{2.35}$$

The second order (lowest) expression for the magnetic field is determined from Maxwell's equation $\nabla \times \bar{E} = -\partial\bar{B}/\partial t$ to be

$$\bar{B}_1 = -\frac{1}{\omega}\nabla \times \bar{E}_0(\bar{r}_0)\sin(\omega t). \tag{2.36}$$

These equations (2.34 to 2.36) can then be substituted into the second order expression of the Lorentz force equation

$$m_e \frac{d\bar{v}_2}{dt} = -q\left[(\delta\bar{r}_1 \cdot \nabla)\bar{E} + v_1 \times \bar{B}_1\right] \tag{2.37}$$

to yield

$$m_e \frac{d\bar{v}_2}{dt} = -\frac{q^2}{m_e\omega^2}\left[\begin{array}{l}\left(\left(\bar{E}_0(\bar{r}_0)\cos(\omega t)\right)\cdot\nabla\right)\bar{E}_0(r)\cos(\omega t) \\ \quad +\left(-\bar{E}_0(\bar{r}_0)\sin(\omega t)\right)\times\left(-\nabla\times\bar{E}_0(\bar{r}_0)\sin(\omega t)\right)\end{array}\right]. \tag{2.38}$$

The sinusoidal $\cos^2(\omega t)$ and $\sin^2(\omega t)$ terms are time-averaged to yield 1/2 and equation 2.38 is rearranged to give

$$m_e\left\langle\frac{d\bar{v}_2}{dt}\right\rangle = -\frac{q^2}{2m_e\omega^2}\left[\left(\bar{E}_0\cdot\nabla\right)\bar{E}_0 + \bar{E}_0\times\left(\nabla\times\bar{E}_0\right)\right]. \tag{2.39}$$

Through the vector identity $\nabla(\bar{A}\cdot\bar{B}) = (\bar{A}\cdot\nabla)\bar{B} + (\bar{B}\cdot\nabla)\bar{A} + \bar{A}\times(\nabla\times\bar{B}) + \bar{B}\times(\nabla\times\bar{A})$, the two terms on the right-hand side of equation 2.39 can be combined into a single term, resulting in the ponderomotive force equation

$$\bar{F}_P = -\frac{q^2}{4m_e\omega^2}\nabla E_0^2 \tag{2.40}$$

and the corresponding ponderomotive potential

$$U_P = \frac{q^2}{4m_e\omega^2}E_0^2. \tag{2.41}$$

From equation 2.41, it is observed that a potential is formed by a spatially varying electromagnetic field distribution. To achieve a large ponderomotive force, which is

necessary for accelerating electrons to high energies, both large field amplitudes and steep gradients are necessary. As described previously, the enhancement and large field gradient provided by an SP wave are ideally suited for the ponderomotive interaction.

2.6 Literature Review

The first experiment to demonstrate that energetic electrons can be generated using SP waves was performed by Zawadzka et al. in 2000 [11]. A schematic of the experimental arrangement is shown in Figure 2.8. The laser pulses used to launched the SP waves were

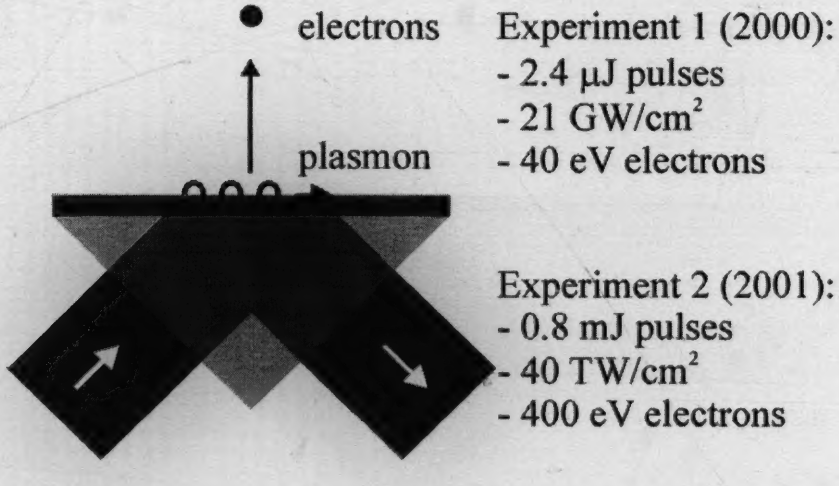

Figure 2.8. Prism coupling method used by Zawadzka et al. [11,12] to launch surface plasmons and accelerate electrons.

generated from a 25 W Argon-ion pumped titanium-sapphire laser oscillator. This system provided 2.4 μJ pulses at a central wavelength of 800 nm and a repetition rate of 250 kHz. The laser beam was directed into a metal-coated prism in the Kretschmann configuration and generated electrons from gold and silver surfaces. Using a time-of-

flight electron energy spectrometer, it was determined that 40 eV electrons were produced from the gold and silver surfaces at a laser intensity of 21 GW/cm^2. Follow-up experiments [12] by the same group showed that even higher energy electrons could be generated using SP waves. In these subsequent experiments, a titanium-sapphire laser amplifier system was implemented, which delivered ~ 0.8 mJ pulses at a repetition rate of 1 kHz. Using the same experimental apparatus, it was shown that electrons with energies ranging up to 400 eV could be generated at a laser intensity of 40 TW/cm^2.

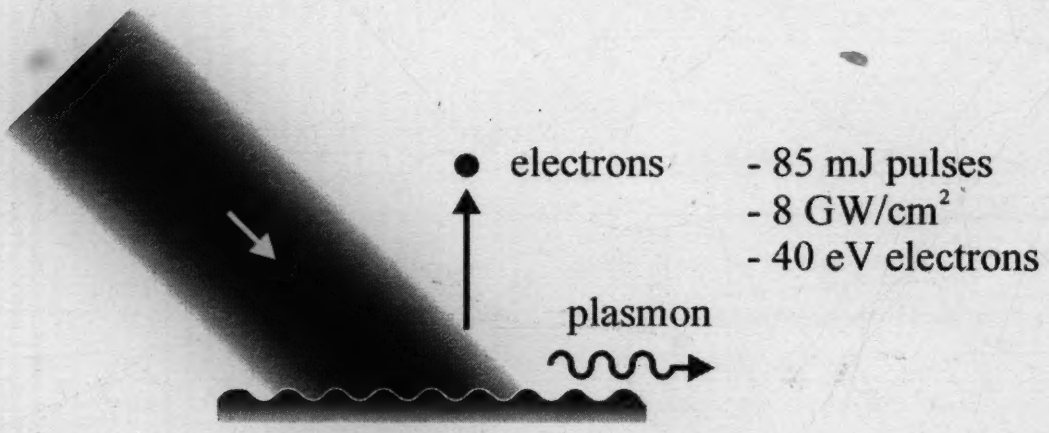

Figure 2.9. Grating coupling method used by Kupersztych et al. [13] to launch surface plasmons and accelerate electrons.

Apart from the aforementioned work, only one other group, located in France, had studied SP enhanced electron acceleration [13]. Instead of the prism coupling method, a diffraction grating, as shown in Figure 2.9, was used to couple laser pulses into SP oscillations on a gold metal film. The laser system consisted of a titanium-sapphire amplifier delivering 85 mJ pulses at a central wavelength of 800 nm. Again, a time-of-flight electron energy measurement apparatus was favored due to the electrons' low energy. At a laser intensity of 8 GW/cm^2, it was shown that 25 eV electrons could be

produced. By increasing the pulse duration to 800 fs, 40 eV electrons were produced; this suggested that the time the electron spent in the laser field is an important parameter.

2.7 Thesis Goals

Essentially only two experiments precede the results presented in this thesis and, at the time, much more work was required to advance this new research area. The specific guiding objectives used to continue the footsteps along the path of development of SP electron acceleration were:

1) To achieve SP electron acceleration using low energy pulses from a titanium-sapphire oscillator,
2) To implement a laser amplifier to study SP electron acceleration at higher intensities than those available from the oscillator,
3) To gain information regarding the electron dynamics at the surface using autocorrelation,
4) To study SP electron acceleration as a function of metal film composition,
5) To track the photocurrent variation over a large range of intensities to investigate the electron emission mechanism,
6) To measure the angular distribution of the photo-accelerated electron packet,
7) To determine the influence of the carrier-envelope phase of the optical waveform on the ponderomotive acceleration process, and
8) To address the issue of ultrashort electron pulse characterization.

One major problem with past research is that no theoretical model existed. During the time that work herein was performed, a noteworthy one-dimensional particle-in-a-cell model was developed for sharp over-dense plasmas [14]. This model, however, focused on ionized plasmas generated by laser ablation, and furthermore, is unable to predict angular spectra or transverse bunching dynamics (both of which are two dimensional effects). Therefore, in addition to the experimental goals, the second major objective of developing a prototype numerical model was added to this project. Realization of such a computer model would, indeed, be extremely useful for visualization of the ponderomotive electron acceleration process and essential for drawing conclusions from experimental data.

2.8 References

[1] N. Peyghambarian, S. W. Koch, and A. Mysyrowicz, *Introduction to Semiconductor Optics*. (Prentice Hall, New Jersey, 1993).

[2] J. R. Reitz, F. J. Milford, and R. W. Christy, *Foundations of Electromagnetic Theory,* 4th Edition (Addison-Wesley, Reading, MA, 1992).

[3] M. E. Caldwell and E. M. Yeatman "Surface-plasmon spatial light modulators based on liquid crystal," *Applied Optics*, vol. **31**, pp. 3880-3891 (1992).

[4] I. Pockrand, "Surface plasma oscillations at silver surfaces with thin transparent and absorbing coatings," *Surface Science*, vol. **72**, pp. 577-588 (1978).

[5] R. A. Millikan, "A Direct Photoelectric Determination of Planck's "h"," *Physical Review*, vol. **7**, pp. 355–388 (1916).

[6] T. Tsang, T. Srinivasan-Rao, and J. Fischer, "Surface-plasmon field-enhanced multiphoton photoelectric emission from metal films," *Physical Review B*, vol. **43**, pp. 8870-8878 (1991).

[7] L. V. Keldysh, "Ionization in the field of a strong electromagnetic wave," *Soviet Physics JETP*, vol. **20**, pp. 1307-1314 (1965).

[8] C. Tóth, G. Farkas, and K. L. Vodopyanov, "Laser-Induced Electron Emission from an Au Surface Irradiated by Single Picosecond Pulses at λ=2.94 μm. The Intermediate Region Between Multiphoton and Tunneling Effects," *Applied Physics B*, vol. **53**, pp. 221-225 (1991).

[9] F. V. Bunkin and M. V. Fedorov, "Cold emission of electrons from the surface of a metal in a strong radiation field," *Soviet Physics JETP*, vol. **21**, pp. 896-899 (1965).

[10] F. F Chen, *Introduction to Plasma Physics and Controlled Fusion*, 2nd Edition (Plenum Press, New York, 1984).

[11] J. Zawadzka, D. Jaroszynski, J. J. Carey, and K. Wynne, "Evanescent-wave acceleration of femtosecond electron bunches," *Nuclear Instruments and Methods in Physics Research A*, vol. **445**, pp. 324-328 (2000).

[12] J. Zawadzka, D. Jaroszynski, J. J. Carey, and K. Wynne, "Evanescent-wave acceleration of ultrashort electron pulses," *Applied Physics Letters*, vol. **79**, pp. 2130-2132 (2001).

[13] J. Kupersztych, P. Monchicourt, and M. Raynaud, "Ponderomotive Acceleration of Photoelectrons in Surface-Plasmon-Assisted Multiphoton Photoelectric Emission," *Physical Review Letters*, vol. **86**, pp. 5180-5183 (2001).

[14] J. Kupersztych, M. Raynaud, and C. Riconda, "Electron acceleration by surface plasma waves in the interaction between femtosecond laser pulses and sharp-edged overdense plasmas," *Physics of Plasmas*, vol. **11**, pp. 1669-1673 (2004).

Chapter 3.0

Theoretical Model

In this chapter, a model for describing SP electron acceleration is derived. The physical description of this model is similar to the simple-man's approach [1], which was originally developed to study ionization and motion of electrons produced from atoms in an intense laser field. In such formalisms, the process occurs via two steps: 1) electrons are stripped from the atom and 2) subsequently interact with the external ionizing laser radiation. An appropriate ionization model is implemented to predict the rate at which electrons are generated, and once freed from the atom, their motion in the laser field is calculated. Similarly, the model described within this thesis is also based on a quasi-classical approach. The electron emission probability is calculated empirically through multiphoton statistics, while their motion in electric and magnetic fields is treated classically via the Lorentz force equation. However, an additional level of complexity has been added to this model due to the fact that the electromagnetic fields cannot be completely described by analytical functions. Instead, the electron trajectories are calculated in response to the electromagnetic fields obtained from rigorous numerical solution of Maxwell's equations. This formalism yields comprehensive information on the dynamics of the SP waves and allows for the investigation of electron acceleration under various situations and SP wave launching conditions.

In what follows, the assumptions and motivations for the various components of the model are described. Specifically, the theoretical description of SP electron acceleration is comprised of two subsidiary models for predicting 1) the electromagnetic wave dynamics of coupling to SP modes and 2) the electron photoemission from a metallic surface. For the first part of the model, the finite-difference time-domain (FDTD) approach is used to solve Maxwell's equations. This method is described in

detail below and is implemented to illustrate the SP field dynamics at a silver metal film surface. Following the light-wave analysis, a description of electron photoemission and its incorporation into the electrodynamic model is discussed. Finally, the combined model is used to visualize ponderomotive electron acceleration in the presence of an SP wave, and predicts the kinetic energy distribution, angular spread, and angle-resolved spectra of the photo-accelerated electron bunch.

A version of this chapter has been published: S. E. Irvine and A. Y. Elezzabi, Physical Review A, 73, 013815, 2006, Copyright (2006) American Physical Society.

3.1 Finite-Difference Time-Domain Numerical Method

Surface plasmon electron acceleration fundamentally arises from the ponderomotive interaction between charged particles and an electromagnetic field gradient. Therefore, the natural and most intuitive first step is to consider the spatial and temporal electromagnetic field distribution of an SP mode confined to the metal-vacuum boundary. To describe the electromagnetics, such as laser pulse propagation and optical-plasmon coupling discussed in Chapter 2 (see Figure 2.4), Maxwell's equations

$$\frac{\partial \vec{H}}{\partial t} = -\frac{1}{\mu_0} \nabla \times \vec{E} \tag{3.1}$$

and

$$\frac{\partial \vec{D}}{\partial t} = \nabla \times \vec{H} \tag{3.2}$$

for the propagation of electromagnetic waves must be solved, where $\vec{H}$ is the magnetic intensity, $\vec{E}$ is the electric field, $\vec{D} = \varepsilon \vec{E}$ is the electric displacement vector, ε is the local permittivity, and μ_0 is the permeability of free space. For the plasmon-based

electron acceleration geometry, an exact analytical solution to equations 3.1 and 3.2 is not possible, as the electric and magnetic fields no longer have harmonic time dependences. Several methods of numerical solution can be applied, however, the finite-difference time-domain (FDTD) method is favored due to its relative simplicity, stability, and full-vector capability.

As already discussed in Chapter 2, only transverse magnetic (TM) modes can couple to SP waves. Therefore, equations 3.1 and 3.2 can be reduced to the following two-dimensional equations for the E_x, E_y, D_x, D_y, and H_z field components:

$$\frac{\partial H_z}{\partial t} = \frac{1}{\mu_0}\left(\frac{\partial E_x}{\partial y} - \frac{\partial E_y}{\partial x}\right), \tag{3.3}$$

$$\frac{\partial D_x}{\partial t} = \frac{\partial H_z}{\partial y}, \tag{3.4}$$

and

$$\frac{\partial D_y}{\partial t} = -\frac{\partial H_z}{\partial x}, \tag{3.5}$$

along with the constitutive relation $\vec{D} = \varepsilon\vec{E}$. Following the treatment in Taflove [2], the space and time derivatives in equations 3.3-3.5 are substituted with second-order centered finite-difference expressions:

$$\frac{D_x\Big|_{i,j}^{n+1/2} - D_x\Big|_{i,j}^{n-1/2}}{\Delta t} = \frac{H_z\Big|_{i,j+1/2}^{n} - H_z\Big|_{i,j-1/2}^{n}}{\Delta y}, \tag{3.6}$$

$$\frac{D_y\Big|_{i,j}^{n+1/2} - D_y\Big|_{i,j}^{n-1/2}}{\Delta t} = -\frac{H_z\Big|_{i+1/2,j}^{n} - H_z\Big|_{i-1/2,j}^{n}}{\Delta x}, \tag{3.7}$$

and

$$\frac{H_z\Big|_{i,j}^{n+1} - H_z\Big|_{i,j}^{n}}{\Delta t} = \frac{1}{\mu_0}\left(\frac{E_x\Big|_{i,j+1/2}^{n+1/2} - E_x\Big|_{i,j-1/2}^{n+1/2}}{\Delta y} - \frac{E_y\Big|_{i+1/2,j}^{n+1/2} - E_y\Big|_{i-1/2,j}^{n+1/2}}{\Delta x}\right), \tag{3.8}$$

where Δx and Δy are the spatial step sizes in the respective x and y directions and Δt is the corresponding temporal step size. Numerically stability is enforced for the condition [2]: $\Delta t \le c^{-1}\left((\Delta x)^{-2} + (\Delta y)^{-2}\right)^{-1/2}$. In equations 3.6-3.8, the $\vec{E}$ and $\vec{H}$ components are separated by half of a grid spacing in both space and time coordinates according to the Yee algorithm [3]. The spatial offsets between the various components is illustrated in Figure 3.1, where it is also indicated that the indices i, j, and n refer to the field value at the spatial coordinates $(i\Delta x, j\Delta y)$ and time $n\Delta t$. The offset of the field components ensures second-order accurate centered expressions for the spatial finite-differences and it implicitly enforces divergence free meshes for both the electric and magnetic fields. It should be noted that $\vec{E}$ and $\vec{D}$ share the same lattice. Furthermore, since the electric and magnetic fields are 'leapfrogged' in time, the finite-difference expressions for the temporal derivatives are also centered and second-order accurate.

For implementation in a computer algorithm, equations 3.6-3.8 are rearranged such that the most advanced time-step appears on the left-hand-side:

$$D_x\Big|_{i,j}^{n+1/2} = D_x\Big|_{i,j}^{n-1/2} + \Delta t\frac{H_z\Big|_{i,j+1/2}^{n} - H_z\Big|_{i,j-1/2}^{n}}{\Delta y}, \tag{3.9}$$

$$D_y\Big|_{i,j}^{n+1/2} = D_y\Big|_{i,j}^{n-1/2} - \Delta t\frac{H_z\Big|_{i+1/2,j}^{n} - H_z\Big|_{i-1/2,j}^{n}}{\Delta x}, \tag{3.10}$$

and

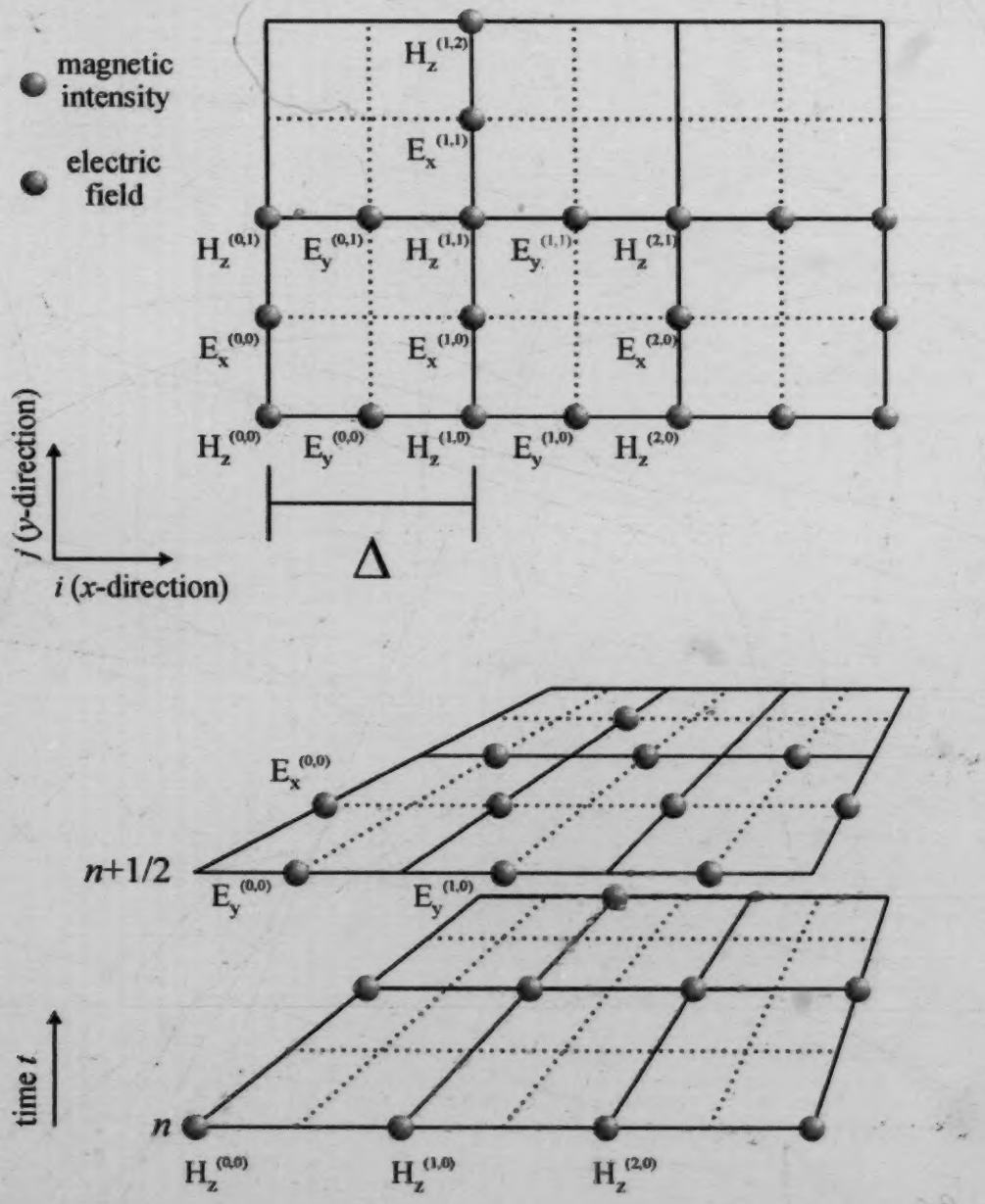

Figure 3.1. (top) Spatial cells of the electric and magnetic field discretization grids, illustrating the relative locations of the lattice points where the E_x, E_y, and H_z field components are calculated. The displacement vector components D_x and D_y share the same points as the corresponding electric field components, however, are omitted for clarity. (bottom) Illustration of the discretization grids along the temporal dimension, indicating that the electric and magnetic fields are also 'leapfrogged' in time as well as in space.

$$H_z\Big|_{i,j}^{n+1} = H_z\Big|_{i,j}^{n} + \frac{\Delta t}{\mu_0}\left(\frac{E_x\Big|_{i,j+1/2}^{n+1/2} - E_x\Big|_{i,j-1/2}^{n+1/2}}{\Delta y} - \frac{E_y\Big|_{i+1/2,j}^{n+1/2} - E_y\Big|_{i-1/2,j}^{n+1/2}}{\Delta x}\right). \tag{3.11}$$

Thus, given the knowledge of the field components at previous times, n and $n+1/2$, the components at the advanced time-steps, $n+1$, can be sequentially determined and the entire spatial and temporal evolution of an electromagnetic interaction can be mapped.

As discussed in Chapter 2, the particular electromagnetic problem under consideration is SP interaction with electrons at a metal film surface. Therefore, a proper description for the response of metallic media is essential for predicting the spatio-temporal evolution of the SP waves. This response is included in the FDTD computation via the Drude theory (derived in Chapter 2) for the frequency-dependent complex dielectric function:

$$\varepsilon_m(\omega) = \varepsilon_0 - \frac{\varepsilon_0\omega_p^2}{\omega(\omega - i\nu_d)}. \tag{3.12}$$

Here, ε_m is incorporated into the model using the auxiliary differential equation formalism [2] in which the constitutive relation $\vec{E} = \varepsilon_m^{-1}(\omega)\vec{D}$ is recast in the time-domain by inverse Fourier transform to yield the supplementary equation

$$\nu_d\frac{d\vec{D}}{dt} + \frac{d^2\vec{D}}{dt^2} = \omega_p^2\varepsilon_0\vec{E} + \nu_d\varepsilon_0\frac{d\vec{E}}{dt} + \varepsilon_0\frac{d^2\vec{E}}{dt^2} \tag{3.13}$$

for the displacement, $\vec{D}$, and electric field, $\vec{E}$, vectors. This equation is also discretized onto the electric field vector mesh and is rearranged to yield:

$$E_x^{n+1} = \frac{(\nu_d\Delta t - 2)D_x^{n+1} + 4D_x^n - (\nu_d\Delta t + 2)D_x^{n-1}}{\varepsilon_0\left(-2 + \nu_d\Delta t - \omega_p^2(\Delta t)^2\right)} \tag{3.14}$$

$$E_y^{n+1} = \frac{(v_d \Delta t - 2) D_y^{n+1} + 4 D_y^n - (v_d \Delta t + 2) D_y^{n-1}}{\varepsilon_0 \left(-2 + v_d \Delta t - \omega_p^2 (\Delta t)^2\right)} \tag{3.15}$$

Thus, the five simultaneous equations (3.9-3.11, 3.14, 3.15) can then be solved numerically for the five components of the $\vec{E}$, $\vec{D}$, and $\vec{H}$ vectors. In locations where the material is a simple dielectric (i.e. not a metal), $\vec{D}$ is replaced with $\varepsilon\vec{E}$ and only the reduced set of three equations (3.9-3.11) is required.

Equations 3.9-3.11 can only update the field values from previous ones, and at some point in the simulation, consideration must be given to the initiation of an electromagnetic wave. One method, known as a 'hard-source', simply assigns a time-varying function to a particular field vector and point on the computational lattice. However, such a stringent assignment does not consider pre-existing local waves and can introduce non-physical reflections similar to a 'hard boundary', the latter of which is discussed below. Alternatively, the total-field scattered-field formulation [2] remains a preferred technique, due to its principal advantage of non-interaction with existing waves that have been scattered by objects in the computational space. A total-field scattered-field boundary, in relation to the computational domain of a typical FDTD simulation, is shown in Figure 3.2a and has vertical indices j_0 on the magnetic intensity lattice and $j_0 - 1/2$ on the electric field lattice. At this boundary, an incident electromagnetic wave propagating in the positive *y*-direction can enter and subsequently interact with objects within the window. This initiation is mediated between three sets of field components [2] that are summed according to the linearity of Maxwell's equations:

$$\vec{E}_{total} = \vec{E}_{incident} + \vec{E}_{scattered}, \tag{3.16}$$

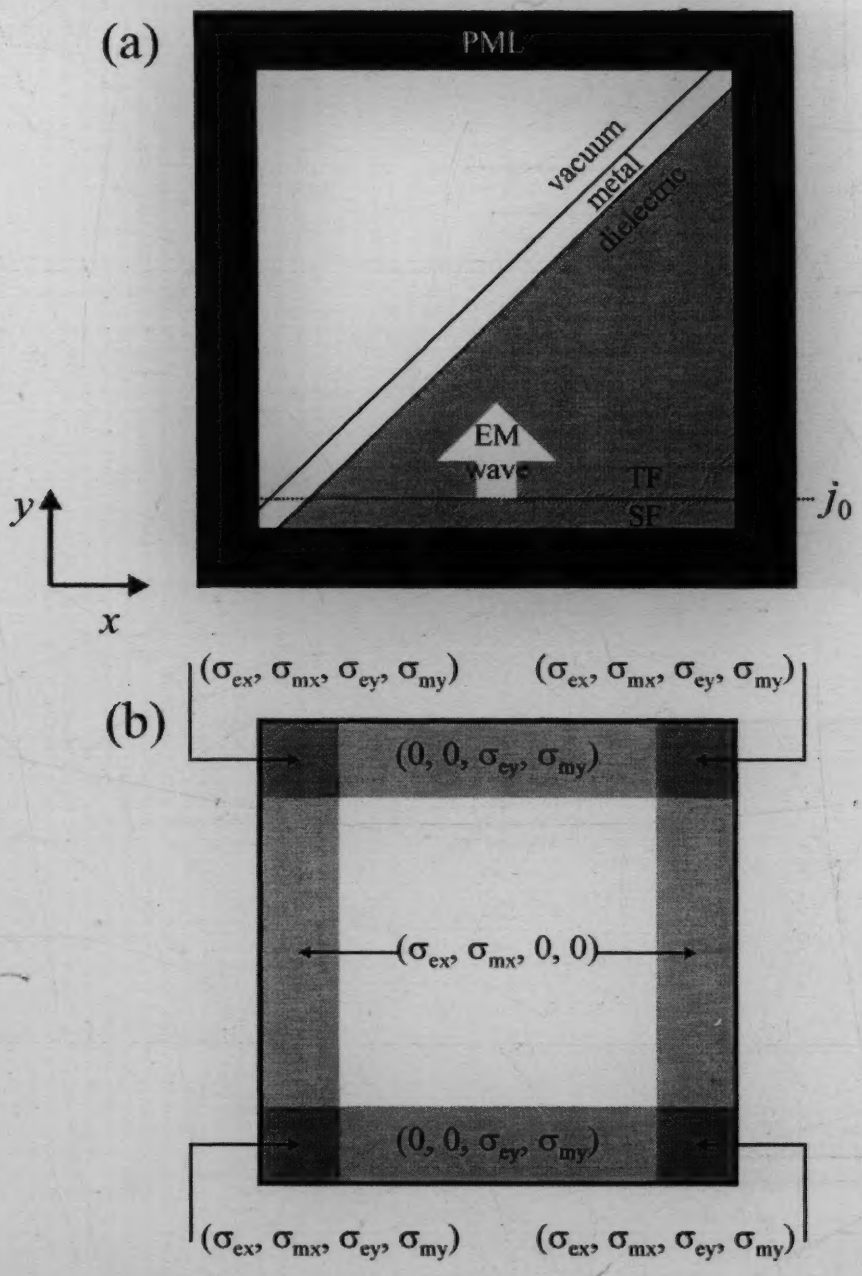

Figure 3.2. (a) Computational domain used for the FDTD portion of the model. The electromagnetic (EM) wave is initiated using the total-field scattered-field (TF SF) technique, and perfectly matched layers (PML) are used to eliminate non-physical reflections from the computational boundaries. (b) PML system used to provide the zero reflection, highly absorbing material to prevent electromagnetic wave reflection at the outermost boundaries of the computational window.

where $\vec{E}_{total}$ is the total electric field, $\vec{E}_{incident}$ is the incident electric field of the electromagnetic wave to be launched, and $\vec{E}_{scattered}$ is the portion of the electric field that is scattered inside the computational window. The same can also be said for the magnetic intensity components:

$$\vec{H}_{total} = \vec{H}_{incident} + \vec{H}_{scattered} \tag{3.17}$$

With such a separation of the fields into the scattered and incident components, equation 3.11 can be written at the total-field scattered-field boundary:

$$\begin{aligned} H_{z,total}\Big|_{i,j_0}^{n+1} = H_{z,total}\Big|_{i,j_0}^{n} \\ + \frac{\Delta t}{\mu_0}\left(\frac{E_{x,total}\Big|_{i,j_0+1/2}^{n+1/2} - E_{x,scattered}\Big|_{i,j_0-1/2}^{n+1/2}}{\Delta y} - \frac{E_{y,total}\Big|_{i+1/2,j_0}^{n+1/2} - E_{y,total}\Big|_{i-1/2,j_0}^{n+1/2}}{\Delta x} \right) \\ + \frac{\Delta t}{\mu_0}\left(\frac{-E_{x,incident}\Big|_{i,j_0-1/2}^{n+1/2}}{\Delta y} \right) \end{aligned} \tag{3.18}$$

Similarly, equations 3.9 is written:

$$\begin{aligned} D_{x,scattered}\Big|_{i,j_0-1/2}^{n+1/2} = D_{x,scattered}\Big|_{i,j_0-1/2}^{n-1/2} + \Delta t \frac{H_{z,total}\Big|_{i,j_0}^{n} - H_{z,scattered}\Big|_{i,j_0-1}^{n}}{\Delta y} \\ - \Delta t \frac{H_{z,incident}\Big|_{i,j_0}^{n}}{\Delta y} \end{aligned}, \tag{3.19}$$

As observed, equations 3.18 and 3.19 are simply equations 3.9 and 3.11 rewritten to include perturbation terms that add the incident electromagnetic field. The time-varying functional form of the incident fields is given by:

$$\begin{pmatrix} \vec{E}_{incident} \\ \vec{H}_{incident} \end{pmatrix} = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix} \cos\left(\omega(t-t_0) + \varphi_{CEP}\right) e^{\left(-\frac{x^2}{w_0^2} - \frac{(t-t_0)^2}{\tau_p^2}\right) 4\ln(2)}, \tag{3.20}$$

where τ_p is the optical pulse duration, w_0 is the spatial width of the beam (full-width at half-maximum), ω is the frequency of the incident radiation, $A_{1,2}$ is the amplitude (either electric and magnetic field), and t_0 is the temporal offset. The additional φ_{CEP} variable, the carrier-envelope phase of the waveform, is a measure of the timing between the peak of the envelope of the pulse and the maximum of the underlying light-oscillation. The influence of this parameter will be discussed in Chapter 5, however, for all other analyses $\varphi_{CEP} = 0$.

In addition to launching an electromagnetic pulse, the interaction of the incident radiation with the outermost boundaries must also be considered. As depicted in Figure 3.2a, only a finite spatial extent can be calculated and the electric and magnetic fields must be truncated at the borders of the computational window. In the simplest case, the field components are forced to be zero at and beyond the edges. However, as is well known, such a 'hard boundary' will also perfectly reflect an incoming wave, which is analogous to electromagnetic wave reflection from an object that is perfectly conducting (e.g. metals). As an alternative to simple truncation, a highly absorbing layer can also be placed adjacent to the boundary to ensure that waves approaching the boundaries are attenuated to negligible amplitude before actually striking and reflecting from the window edges. The most successful and widely used method to achieve this is the Berenger [4] perfectly matched layer (PML). Here, the electric and magnetic losses are matched and provide a highly absorbing, yet virtually non-reflective medium.

The Berenger perfectly matched layer is discussed in detail elsewhere [2,4], however, its main features will be highlighted here. First, consider Maxwell's equations describing TM mode propagation in a lossy medium:

$$\varepsilon_0 \frac{\partial E_x}{\partial t} + \sigma_e E_x = \frac{\partial H_z}{\partial y}, \tag{3.21}$$

$$\varepsilon_0 \frac{\partial E_y}{\partial t} + \sigma_e E_y = -\frac{\partial H_z}{\partial x}, \tag{3.22}$$

and

$$\mu_0 \frac{\partial H_z}{\partial t} + \sigma_m H_z = \left(\frac{\partial E_x}{\partial y} - \frac{\partial E_y}{\partial x} \right), \tag{3.23}$$

where σ_e and σ_m are the electric and magnetic conductivities, respectively, and it is assumed that $\vec{D} = \varepsilon_0 \vec{E}$. Zero reflection from an interface between vacuum and the lossy medium can be achieved if the impedance matching condition $\sigma_e / \varepsilon_0 = \sigma_m / \mu_0$ is satisfied, however, only for the special case of normal incidence [5]. As an alternative to this lossy material, PML media is implemented. The principle advantage of the PML technique is the splitting of the magnetic intensity in equations 3.21-3.23 into two separate components, H_{zx} and H_{zy}, yielding a new set of four equations rather than three:

$$\varepsilon_0 \frac{\partial E_x}{\partial t} + \sigma_{ey} E_x = \frac{\partial \left(H_{zx} + H_{zy} \right)}{\partial y}, \tag{3.24}$$

$$\varepsilon_0 \frac{\partial E_y}{\partial t} + \sigma_{ey} E_y = -\frac{\partial \left(H_{zx} + H_{zy} \right)}{\partial x}, \tag{3.25}$$

$$\mu_0 \frac{\partial H_{zx}}{\partial t} + \sigma_{mx} H_{zx} = -\frac{\partial E_y}{\partial x}, \tag{3.26}$$

and

$$\mu_0 \frac{\partial H_{zy}}{\partial t} + \sigma_{my} H_{zy} = \frac{\partial E_x}{\partial y}, \tag{3.27}$$

where additional electric (σ_{ex}, σ_{ey}) and magnetic (σ_{mx}, σ_{my}) conductivities are permitted for each direction. Notice that wave propagation in a PML medium along the *x*-direction is governed only by equations 3.25 and 3.26, while wave propagation along the *y*-direction is governed only by equations 3.24 and 3.27 (also signified by the additional *x*, *y* indices in the magnetic intensity). Therefore, absorption in each direction can be independently varied, i.e. the medium can be made to absorb along only one direction, both, or neither (vacuum). It is this added degree of freedom that allows for the implementation of PML media in the FDTD simulation to form highly absorbing layers that have zero reflection at *any* angle of incidence. By deriving Fresnel's reflection coefficients for the split set of Maxwell's equations 3.24-3.27 [4], it can be shown that a reflection coefficient of zero between two PML media can be achieved for:

1) interfaces normal to the *x*-direction which have the condition of identical σ_{ey} and σ_{my}

and

2) interfaces normal to the *y*-direction which have the condition of identical σ_{ex} and σ_{mx}.

Thus, the two-dimensional system of PML layers shown in Figure 3.2b can be used to eliminate reflections, at any angle of incidence, from the computational window boundaries.

3.2 Electron Photoemission and Motion

Consideration must be also given to the photo-ejection mechanism of the conduction-band electrons of the metal film in the presence of the laser excitation, as the final

ponderomotive energy gain is a strong function of the electron's initial position with respect to the accelerating electric field of the SP, E_{SP}. To incorporate electron emission into the model, test electrons are placed uniformly along the length of the film at several instances in time to represent all possible trajectories of any electron that is liberated from the film surface. The rate of photoelectron generation, however, is not identical at all spatial and temporal coordinates, i.e. more electrons are emitted at the peak of the laser pulse relative to the wings. Therefore, relative weights are assigned to each sample electron trajectory that scale according to the local intensity of the optical pulse $I_{laser}^{m}(x,y,t)$, where m is the order of the emission process. Since the experimental measurements (see Chapter 4) will focus on the excitation of silver films using $\lambda_0 = 800$ nm laser radiation, m would be equal to 3 in the multiphoton regime ($\gamma > 1$). For experiments using higher intensities, m will be reduced to values near 1.5, corresponding to Keldysh emission. In such a case, the photoemission probability cannot be represented as a simple power-law scaling of the intensity. Rather, the interpolated photocurrent-intensity characteristic is employed to yield the most accurate results as will be demonstrated.

Once freed from the metal surface, electrons are acted upon by the electric and magnetic fields of the SP wave. The nature of this interaction is governed by the classical Lorentz force equation:

$$\frac{d\vec{v}}{dt} = \frac{q}{m_e}\left(\vec{E} + \mu_0 \vec{v} \times \vec{H}\right), \tag{3.28}$$

where q/m_e and $\vec{v}$ are the charge-to-mass ratio and velocity of the electron, respectively. Using bilinear interpolation [6] for the electric and magnetic fields

components, equation 3.28 is numerically solved for the velocity and position of each test electron. Any electron trajectory that has traced its path back to the vacuum-metal boundary is assumed to be absorbed and secondary electron emission is ignored.

Since this model does not take into account electron-electron interaction, the analysis and discussion are restricted to cases in which the peak current density does not exceed space-charge saturation. The upper limit for the peak current density can be estimated from the Child-Langmuir law [7,8]: $J_{peak} = (4/9)\varepsilon_0\sqrt{2q/m_e}\left(V^{3/2}/d_{acc}^2\right)$, where V is the potential through which the electrons are extracted and d_{acc} is the distance over which the potential is applied. Using experimental values for the ponderomotive potential of >100 eV and an acceleration distance of < 1 μm, a space-charge-limited peak current density of $>10^5$ A/cm^2 is determined. This value is much larger than the typical values relevant to the experiments described in Chapter 4 ($<10^3$ A/cm^2), and therefore, the model can be applied in these cases.

3.3 Model Results

To closely resemble actual experimental conditions, the metal film parameters are taken to be those of a silver metal film: d =50 nm, ω_p =5.73×10^{15} Hz, and ν_d =1.3×10^{14} Hz [9]. For optimum coupling at θ_{SP} =45°, a value of n_{prism} =1.6 is used for λ_0 =800 nm radiation, which agrees well with the calculated value. The spatial step sizes of the computational lattice are chosen to be $\Delta x = \Delta y = d/10 = 5$ nm with a corresponding temporal step size of $\Delta t = 5\times10^{-3}$ fs.

Figure 3.3a illustrates the total electric field distribution of an SP that has been launched using a Gaussian excitation pulse having a duration of τ_p=30 fs. For a perfectly smooth silver film, an enhancement factor η=3.7 of the electric field is observed at the surface. The electric field decays evanescently away from the silver surface and into vacuum with α_0^{-1}= 240 nm, which is in excellent agreement with the value of 243 nm calculated from equation 2.23: $\alpha_0 = \frac{\omega}{c}\sqrt{n_{prism}^2 \sin^2(\theta_{SP}) - 1}$. A vector representation of a segment of the electric field near the metallic surface is shown in Figure 3.3b. Electric field lines originate and terminate on the silver film surface, indicating regions of positive and negative charge (also shown in Figure 3.3b) commensurate with the SP wave. As expected, these local charges oscillate at a frequency equal to that of the incident laser radiation (375 THz).

Once the electric and magnetic fields of the SP wave have been calculated, the second stage of the model is implemented in which electron emission from the thin silver surface is considered. Figure 3.4 demonstrates the complex motion of an illustrative set of 20 electron trajectories that have been placed at the peak amplitude of the SP wave. Due to the electric field gradient, each test electron experiences asymmetric forces during successive cycles of the SP wave, thus leading to a preferential displacement. The oscillating nature of the SP wave is clearly evidenced by the 'quivering' motion of each sample electron trajectory. A typical distance between inflection points of the most energetic electron in the sample set at E_{SP}=2.7×10^9 V/cm is calculated to be 36 nm, suggesting that the effective ponderomotive force acts on a few hundred nanometer

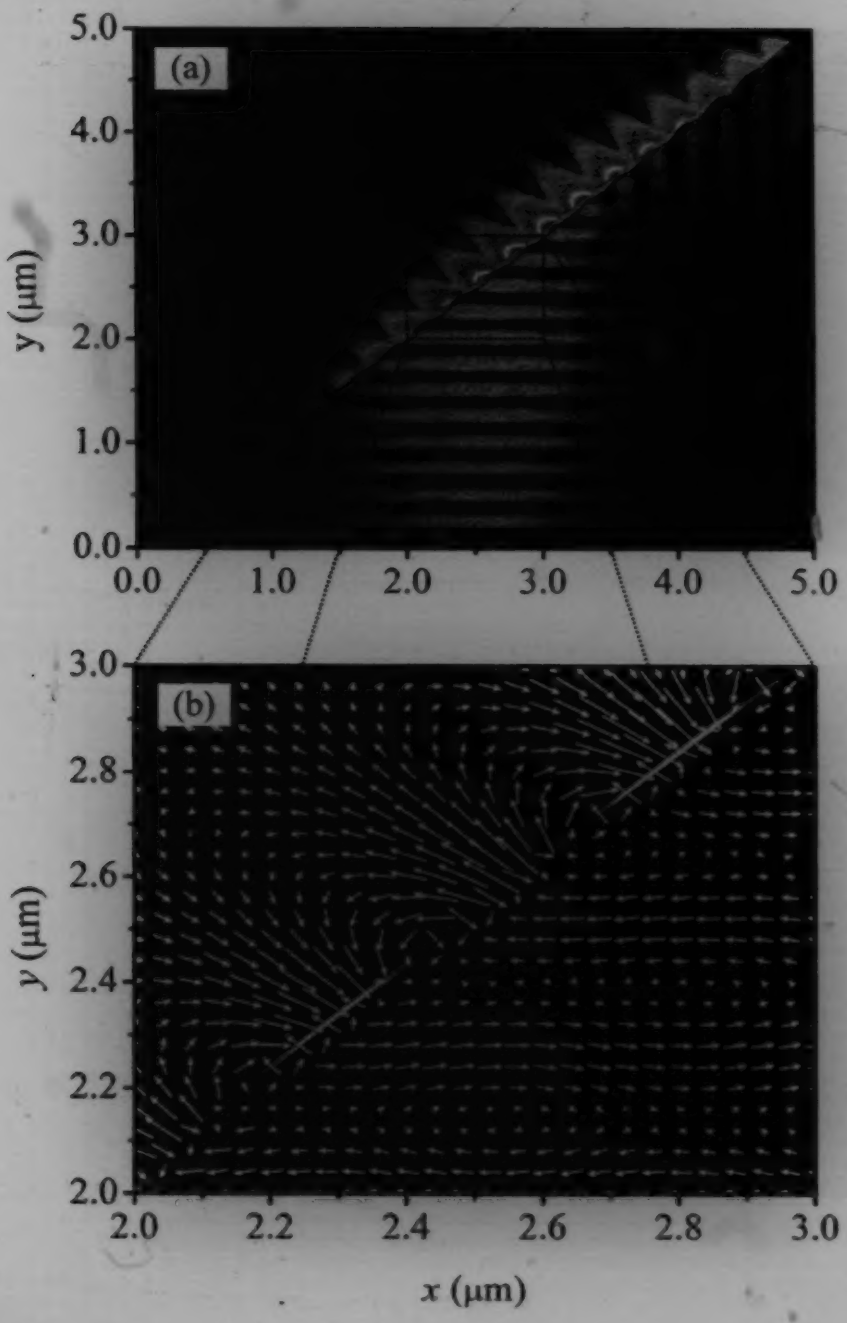

Figure 3.3. (a) Distribution of the total electric field during the excitation of the SP wave. An enhanced electric field is observed at the surface which decays into vacuum with a characteristic length of α_0^{-1}=240 nm. (b) Vector representation of the electric field distribution near the silver film surface, which is overlaid with the local charge density.

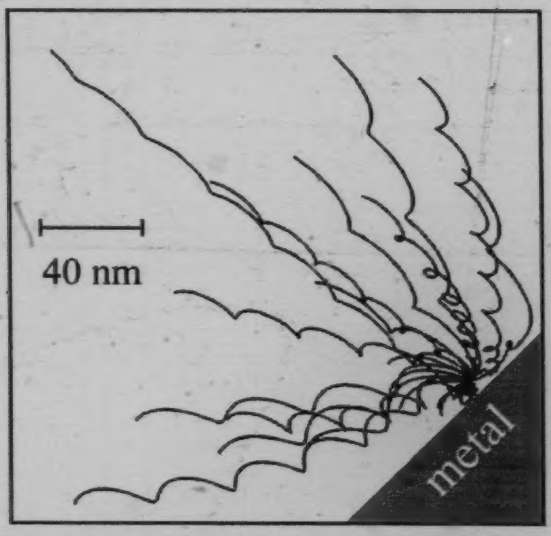

Figure 3.4. A sample set of 20 electron trajectories are shown. The quivering motion of the electrons in the oscillating electromagnetic field of the SP is evident with a distance of ~36 nm between the inflection points of the most energetic electrons.

spatial scale. The multiple inflection points indicate that the electrons are dephased with the propagating SP, thus precluding linear acceleration.

Calculation of kinetic energy spectra requires the number of trajectories to be increased by orders of magnitude to sample the entire SP wave along its spatial and temporal extent. Figure 3.5 illustrates three representative kinetic energy spectra using 10^6 test electron trajectories and a multiphoton order of $m=3$. The energy distribution curves exhibit the same overall character, with progressively higher kinetic energy values for increasing E_{SP}. As electrons are emitted over the spatial and temporal extent of the optical pulse, they experience a wide range of intensities and phases. Consequently, relatively broad kinetic energy spectra are expected. For the typical experimental electric field value of 2.7×10^9 V/cm [10], the calculated average kinetic energy and full-width at half-maximum of the spectrum are 1.06 and 1.04 keV, respectively. When the emission

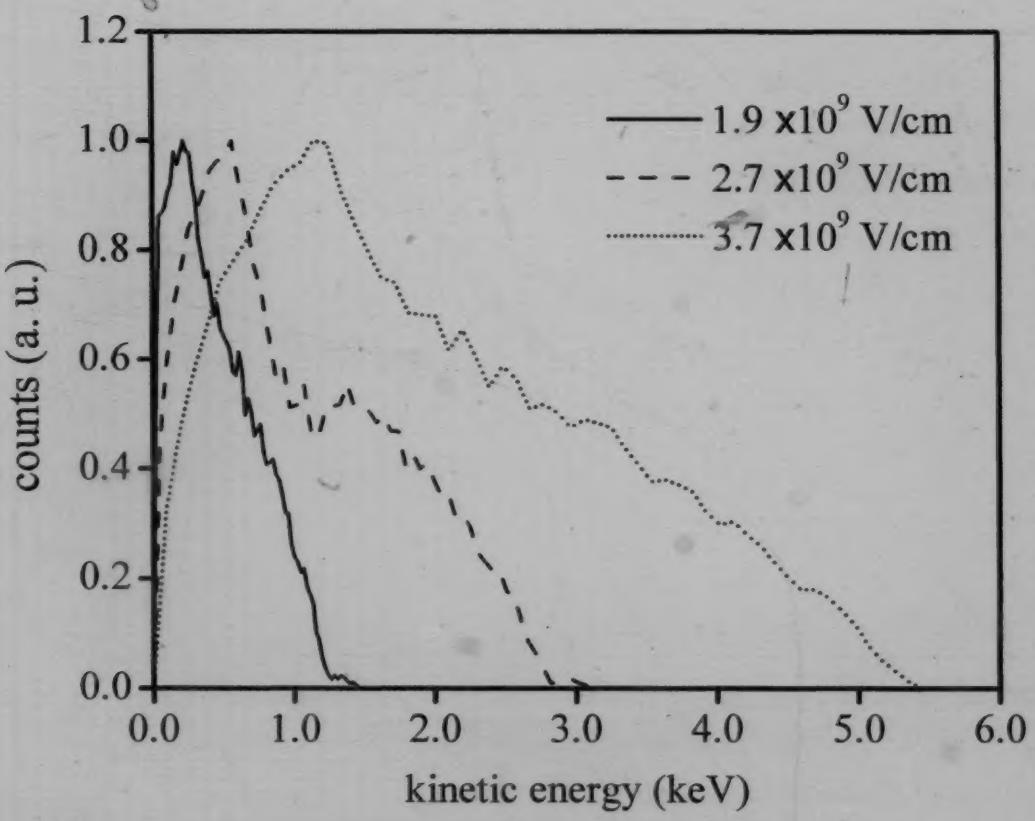

Figure 3.5. Kinetic energy distribution of the SP accelerated electrons for E_{SP} values of 1.9×10^9, 2.7×10^9, and 3.7×10^9 V/cm.

region is confined to within the peak of the pulse (i.e. within 10% of the spatial full-width at half-maximum of the optical pulse) much narrower experimental and calculated spectra are observed. Such a confinement can arise from the particular surface phology (surface roughness) of the metal film [11,12] and is discussed in detail in hapter 4.

The maximum energies of the electron pulses are highly dependent on the magnitude of the surface electric field, E_{SP}. Specifically, the ponderomotive potential, $U_p = (q^2 E_{SP}^2)/(4m_e\omega^2)$, indicates that the final kinetic energy should scale quadratically with electric field. To investigate this dependence, the maxima of several energy spectra are calculated for various E_{SP} values and are illustrated in Figure 3.6. The expected quadratic dependence predicted from the ponderomotive equation is not satisfied, and instead, a higher-order dependence of 2.6 is calculated. The discrepancy arises as the ponderomotive potential equation is derived for an electromagnetic wave of infinite duration, which is in contrast to the effective ponderomotive potential calculated for few-cycle pulses. For low values of E_{SP} ($<10^8$ V/cm) electrons spend a significant amount of time in the field of the SP, which can be comparable to the plasmon's lifetime and essentially E_{SP} 'turns off' before the ponderomotive potential can be completely converted into electron kinetic energy. If, on the other hand, E_{SP} is large ($>10^9$ V/cm), the electrons are pushed out of the field within the interval of the laser pulse. In such a case, significant transfer of energy from ponderomotive potential to kinetic occurs in a duration less than the duration of the laser pulse. An oversimplified analysis that incorporates temporal information in the ponderomotive equation can be derived and

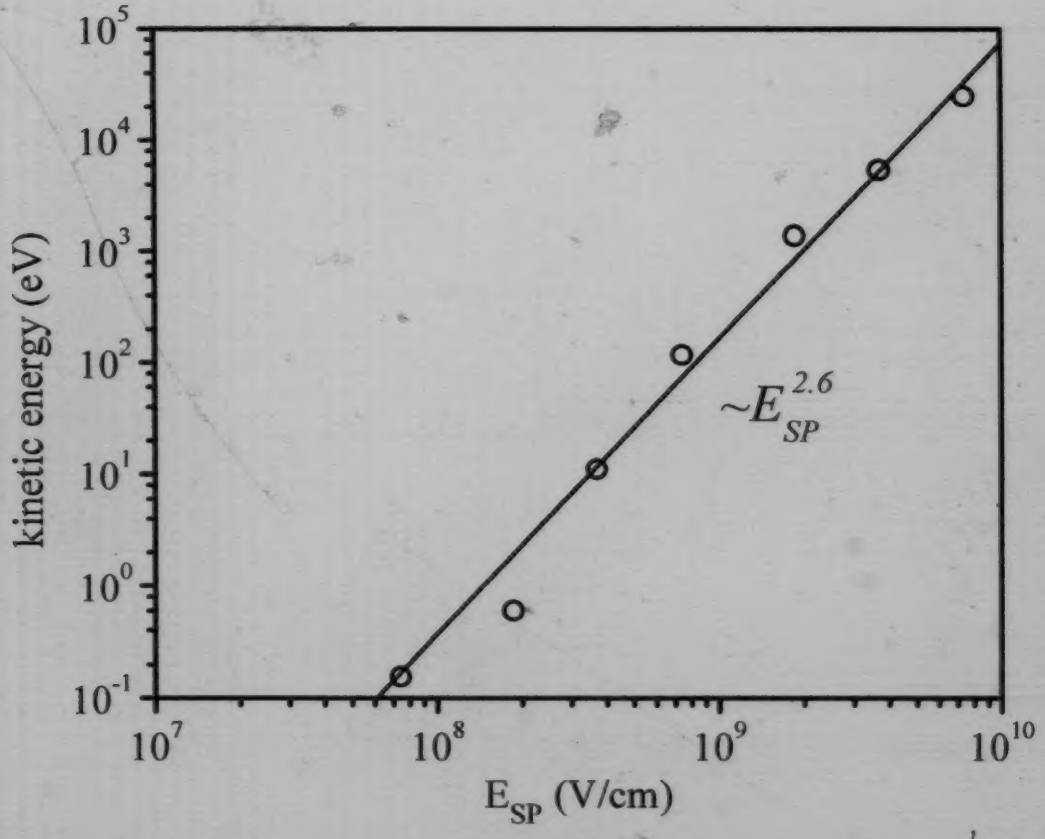

Figure 3.6. Calculated variation of maximum observed kinetic energy with E_{sp} (circles) for a τ_p=30 fs pulse. The linear fit (solid line) indicates a 2.6 power dependence.

used to illustrate this point. The ponderomotive potential as a function of distance for the evanescent SP field can be expressed as:

$$U_p = \frac{q^2 E_{SP}^2}{4m_e\omega^2} e^{-2\alpha_0 z}, \tag{3.29}$$

from which the corresponding force equation can be obtained:

$$\frac{dv_z}{dt} = \frac{\alpha_0 q^2 E_{SP}^2}{2m_e^2\omega^2} e^{-2\alpha_0 z} \tag{3.30}$$

where $v_z = dz/dt$ is the velocity, z is the position, and t is time. To incorporate the effect of a finite duration optical pulse, equation 3.30 is numerically integrated over the interval $t = (0, \tau_p)$, where τ_p is the laser pulse duration. For τ_p=30 fs and α_0^{-1}=240 nm, Figure 3.7 shows the variation of maximum kinetic energy with E_{SP}. At electric fields above 4.5×10^9 V/cm, the dependence is second order as expected from the ponderomotive potential equation. However, for electric fields below 4.0×10^8 V/cm, the final kinetic energy depends on the fourth power of electric field. To further analyze such power-dependences, the duration of the optical excitation pulse in the model calculations is increased to ~1 ps while maintaining the electric field amplitude. Results of this are shown in Figure 3.8, which predict an order of 2.4 for the electric field dependence. Clearly, the model results in Figure 3.6 indicate that the electron kinetic energy dependence on electric field lies in an intermediate regime.

The precise time of photoemission, in relation to the phase of the optical wave, is also a significant parameter that influences the final energy of an electron. Simple analyses, such as the one described above, do not include any information regarding the phase of the SP wave during electron ejection, and therefore, are unable to predict

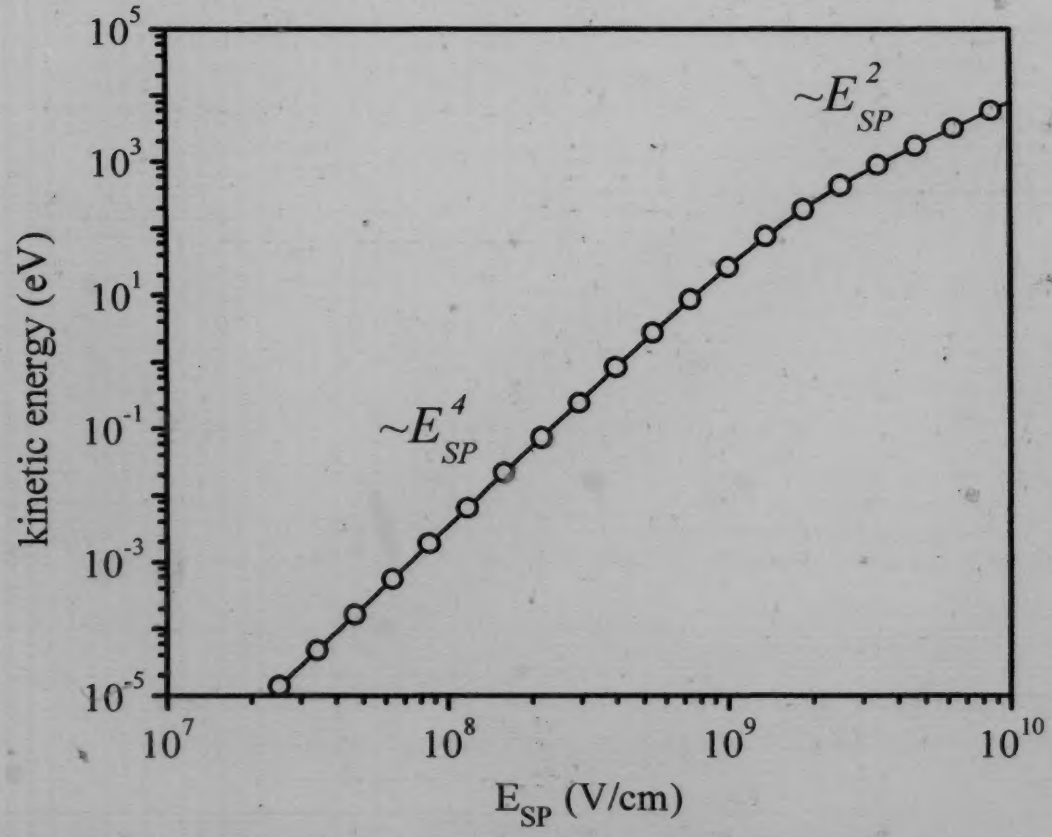

Figure 3.7. A simple scaling relation between maximum kinetic energy and electric field for electrons ponderomotively accelerated with an ultrashort pulse with duration of τ_p=30 fs and α_0^{-1}=240 nm. For electric fields below 4.0×10^8 V/cm, a fourth order power dependence is observed while a second order power dependence exists at higher field strengths.

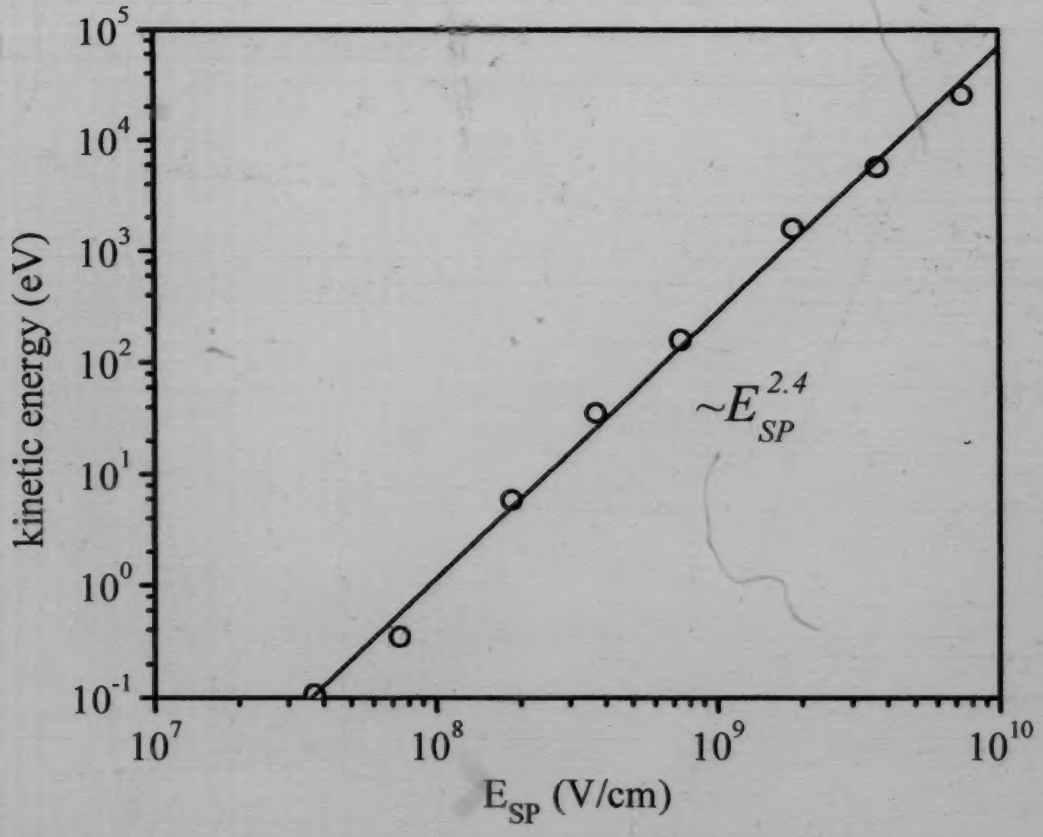

Figure 3.8. Variation of maximum observed kinetic energy with E_{SP} for a ~1 ps excitation pulse, calculated from the FDTD model, indicating a 2.4 power dependence.

resulting effects on ponderomotive energy gain. It is well known [13,14] that the initial phase of the electron, with respect to the oscillating electromagnetic field, can lead to a substantial increase (or decrease) of the maximum observed kinetic energy. These phase effects can be incorporated in the ponderomotive equation through an effective parameter:

$$U_p = \beta \frac{q^2 E_{SP}^2}{4m\omega^2}, \tag{3.31}$$

where β accounts for additional energy gain/loss due to the initial electron location in relation to the phase of E_{SP} at the time of photoemission. This prefactor can range from 0 (i.e. the ponderomotive force is exactly cancelled) to some value greater than unity. Through comparison of Figures 3.6 and 3.7 for E_{SP} >2.5×10^9 V/cm, it is determined that $\beta \cong 6$. A straightforward one-dimensional analysis [14] reveals that the electron can gain energies ranging from 0 to 8 times U_p, which is in good agreement with the value of $\beta \cong 6$ determined from the simple analytical model.

A representative illustration capturing both the electron motion and the accelerating SP electric field can be obtained by overlaying the individual weighted test electrons on the electromagnetic field distribution. Several snap-shots at times ranging from -20 fs to 180 fs are shown in Figure 3.9. As the laser radiation impinges on the silver film, electrons are released into the SP field, with maximum emission occurring at the peak of the laser pulse ($t = 0$ fs). After the decay of the SP wave, the electrons continue propagating and disperse due to their broad kinetic energy distribution. Examination of the snap-shot at ~60 fs in Figure 3.9 reveals several periodic regions of local maximum and minimum electron concentrations. In particular, it is determined that

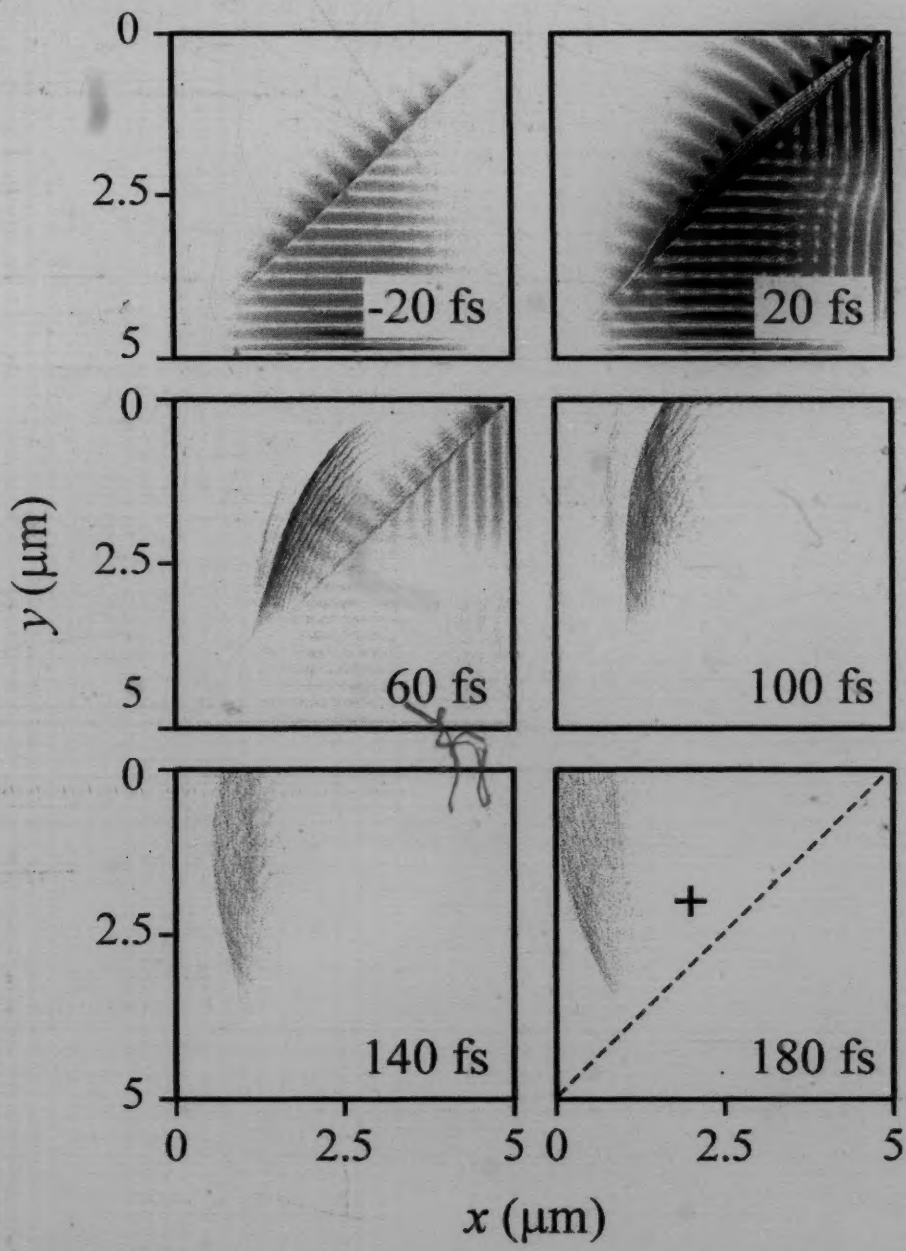

Figure 3.9. Snap-shots at times of -20, 20, 60, 100, 140, and 180 fs illustrate the behavior of electrons under the influence of the SP field. Time zero indicates when the center of the excitation pulse reaches the silver film. The marker in the final frame (180 fs) designates the location of electron number distribution sampling in relation to the metal surface (dashed line). The results of the electron number distribution sampling are shown in Figure 3.10.

approximately 14 electron packets, corresponding to the number of electric field oscillations within the 30 fs laser pulse, have developed across the spatial extent of the SP wave. Such dynamics are an indication that microbunching has occurred.

The femtosecond electron microbunching can be more readily observed by sampling the electron number distribution at a particular region adjacent to the silver film. Figure 3.10 depicts the temporal evolution of the electron number distribution at a point ~140 nm away from the surface of the silver film (indicated by cross-hairs in Figure 3.9). The overall envelope of the curve resembles a fast-rise (16 fs) and an exponential decay function with a peak value at 70 fs and a $1/e$ point located ~20 fs later. Within this envelope are ~14 nearly equally spaced individual peaks corresponding to the number of cycles in the optical excitation pulse. The average duration of the individual electron bunches is determined to be ~4 fs, suggesting a possible route for generating sub-τ_p ultrashort bunches of electrons using SP waves.

The spatial emission profile of the photo-accelerated electrons is another important characteristic that can be studied using this model. Shown in Figure 3.11 is the calculated in-plane angular distribution of electrons subjected to an E_{SP}=2.7×10^9 V/cm. It is observed that the plasmon-accelerated electrons are highly directional, with the most favorable acceleration direction along 18.2° away from the normal of the metal surface. Inherent to the SP coupling geometry is an obvious asymmetry; the SP wave must propagate along the silver film surface in the direction that conserves momentum. It is surmised that the SP wave effectively 'drags' the electrons slightly along its direction of propagation, resulting in an angular distribution whose maximum is tilted along the k_x-direction. Notably, similar angular distributions have been observed experimentally [10].

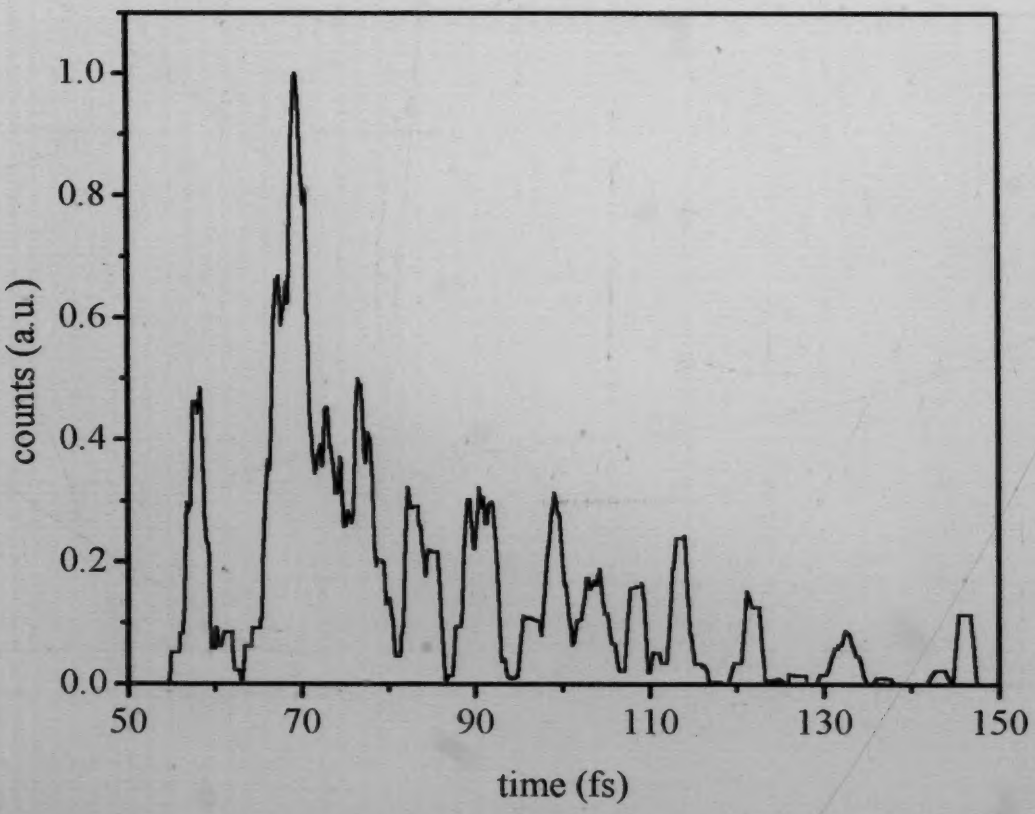

Figure 3.10. Sampled electron number distribution as a function of time for a fixed location ~140 nm in front of the silver film surface (indicated in Figure 3.9 by a cross-hair). The effect of microbunching is evident as several individual ~ 4 fs peaks have developed within the envelope.

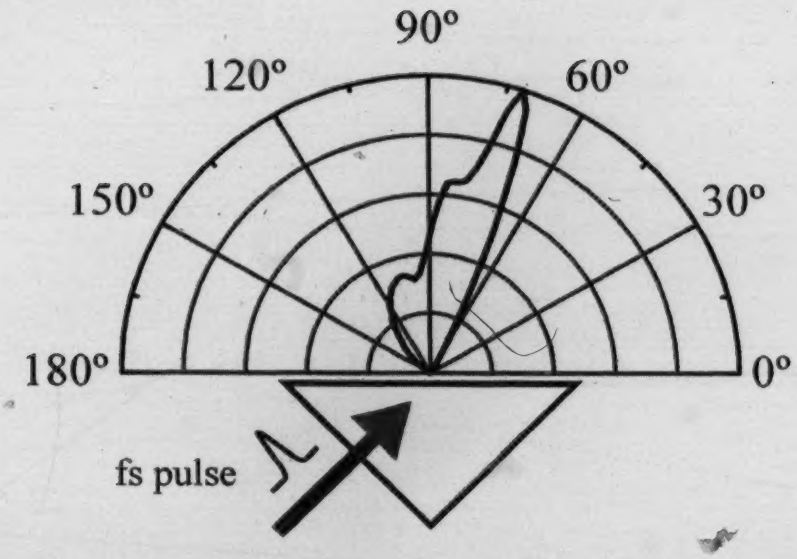

Figure 3.11. In-plane angular distribution of the photo-accelerated electrons. The peak emission angle occurs at 18.2° away from the normal to the surface.

So far, the model has been employed to resolve the number of electrons as a function of energy, and then as a function of angle. Angle-resolved energy distributions will provide greater insight into the physics underlying the plasmon-assisted electron acceleration mechanism. Figure 3.12 illustrates the angle-resolved energy distributions and their dependence on the specific value of E_{SP}. For the lowest field of E_{SP}=3.7×10^8 V/cm, the distribution spans the complete range from 0 to 180°, with the highest energy electrons near 75°. Lateral distributions appearing at angles near 0° and 180° indicate that a significant number of electrons are accelerated along the silver film surface. The lateral distributions vanish for E_{SP} beyond 7.4×10^8 V/cm and the vast majority of electrons are accelerated away from the film surface. Within each of the asymmetric angle-resolved energy distributions, one observes non-uniform regions at particular angles. Tailoring the kinetic energy spectra can be achieved by selecting these particular angles, which can be

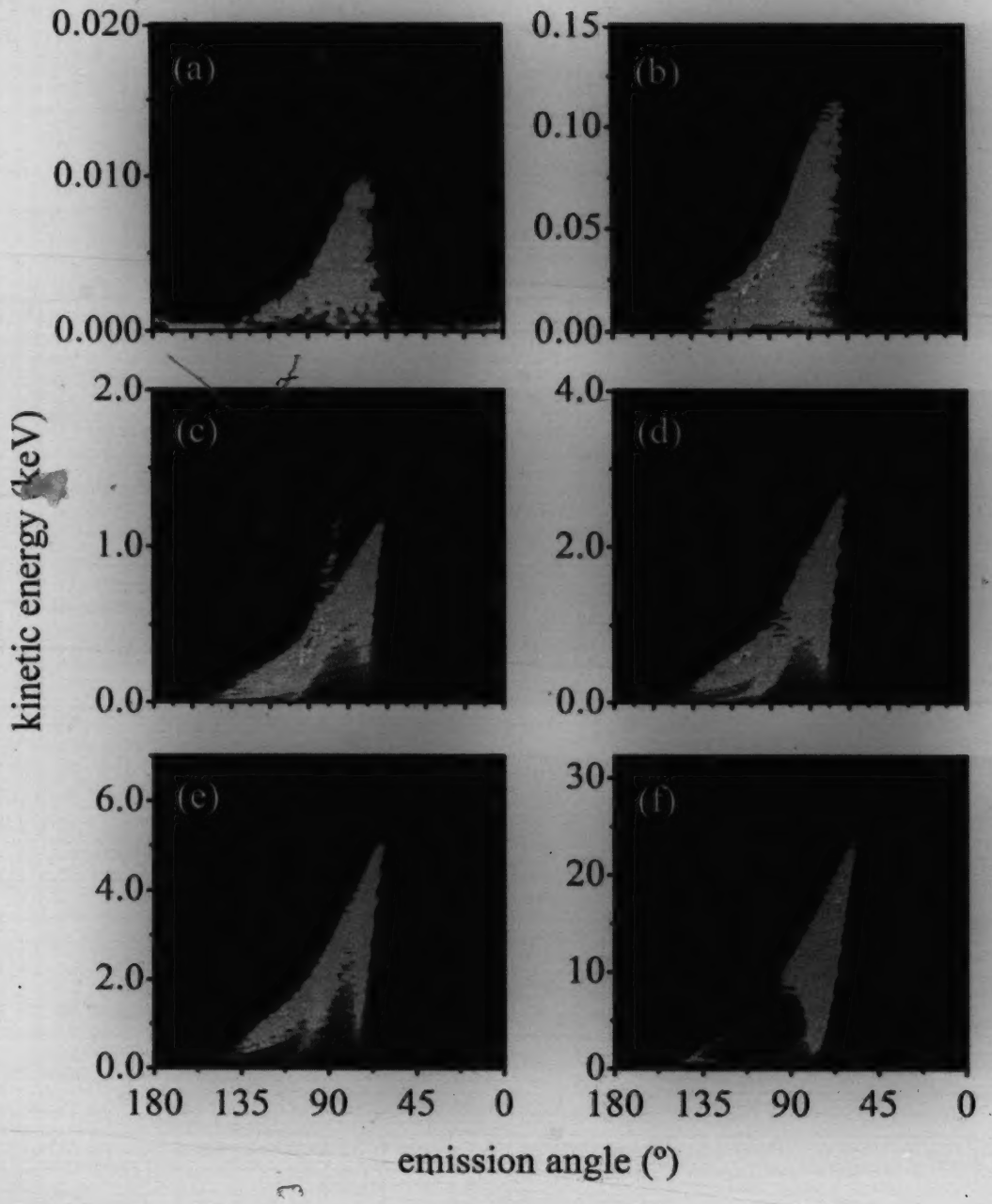

Figure 3.12. Angle-resolved energy distributions for several E_{SP} of: a) 3.7×10^{8}, b) 7.4×10^{8}, c) 1.9×10^{9}, d) 2.7×10^{9}, e) 3.7×10^{9}, and f) 7.4×10^{9} V/cm.

realized experimentally by placing an aperture in close proximity to the emission region of the metal film. To illustrate the ability to select specific energies, various slices along three different panels of Figure 3.12 are taken and shown in Figure 3.13. For the lowest field strength of E_{SP}=3.7×10^8 V/cm (Figure 3.13a), lateral slices are taken at three representative angles of 5°, 90° and 175°. Selected kinetic energy distributions along two larger E_{SP} values of 2.7×10^9 V/cm and 7.4×10^9 V/cm are shown in Figures 3.13b and 3.13c, respectively. Indeed, the selectivity and control over the shape and maximum energy of the spectra is evident. It should be emphasized that for E_{SP}=7.4×10^9 V/cm and for a distribution along 90°, a highly peaked (~ 10 keV) kinetic energy spectrum is apparent. Such results show that it is feasible to produce pseudo-narrowband electron pulses having high energies.

3.4 Summary

A novel quasi-classical model for describing SP electron acceleration was derived and discussed in this Chapter. The model is based on FDTD solution of Maxwell's equations and includes the non-linear electron photoemission characteristics of metallic surfaces. Using realistic experimental parameters, it is established that electrons can be effectively accelerated to high-energy within several 10's of femtoseconds. When the electron kinetic energy dependence on electric field was investigated, it was revealed that the spatial extent of the SP wave and the time the electrons spend in the ponderomotive potential influences the electron energy. Femtosecond electron microbunching was also observed in the spatially resolved electron trajectories. By temporally sampling the electron number distribution at a particular region adjacent to the metal film, bunch

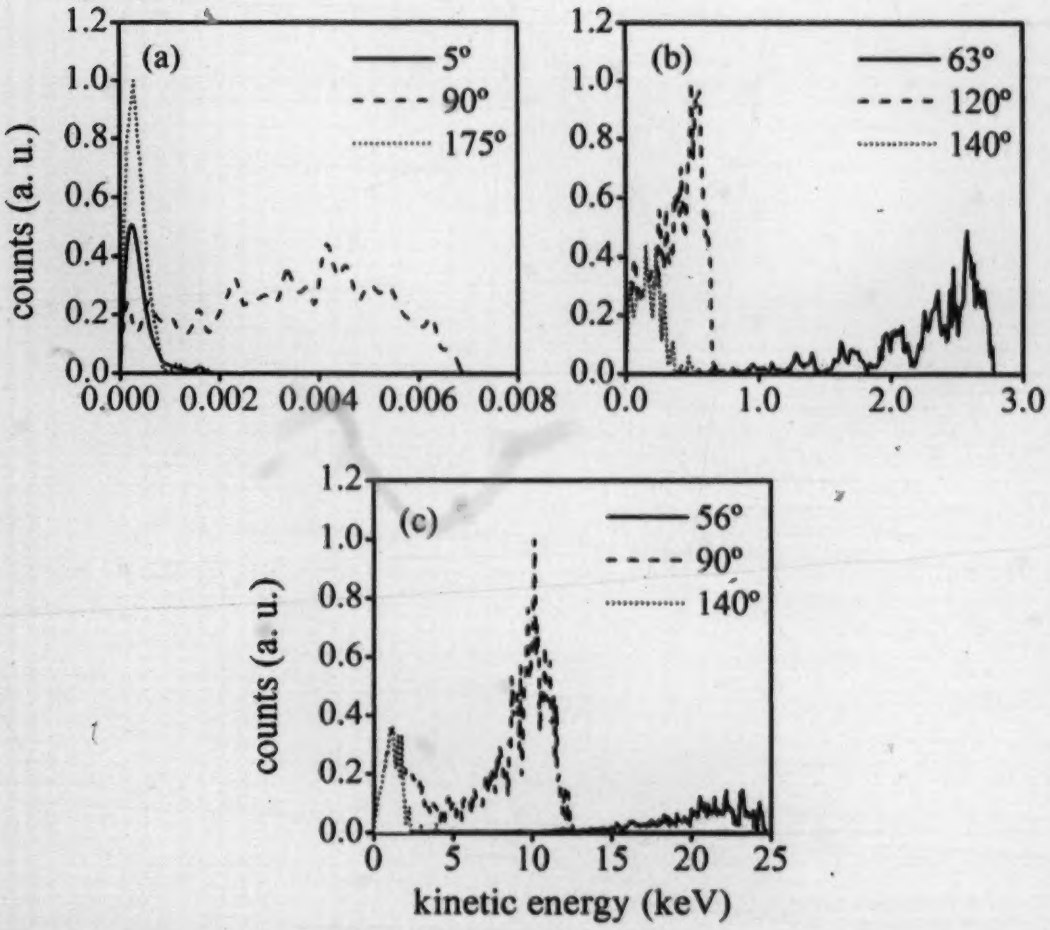

Figure 3.13. Selected slices along three panels of Figure 3.12 corresponding to a) 3.7×10^{8}, b) 2.7×10^{9}, and c) 7.4×10^{9} V/cm. These energy distributions demonstrate the selectivity and control over the shape and maximum energy of the spectra.

lengths of few femtoseconds are detected. Angle-resolved kinetic energy spectra also illustrate that the kinetic energy distributions can be modified to yield quasi-narrowband spectra.

3.5 References

[1] H. B. van Linden van den Heuvell and H. G. Muller, "Limiting Cases of Excess-Photon Ionization," in *Cambridge Studies in Modern Optics 8: Multiphoton Processes*, edited by S. J. Smith and P. L. Knight (Cambridge University Press, Cambridge, 1988), pp. 25-34.

[2] A. Taflove, *Computational Electrodynamics* (Artech House, Boston, 1995).

[3] K. S. Yee, "Numerical Solution of Initial Boundary Value Problems Involving Maxwells Equations in Isotropic Media," *IEEE Transactions on Antennas and Propagation*, vol. **14**, pp. 302-307 (1966).

[4] J. P. Berenger, "A Perfectly Matched Layer for the Absorption of Electromagnetic Waves," *Journal of Computational Physics,* vol. **114**, pp. 185-200 (1994).

[5] R. Holland and J. W. Williams, "Total-field versus scattered-field finite-difference codes: A comparative assessment," *IEEE Transactions on Nuclear Science*, vol. **30**, pp. 4583-4588 (1983).

[6] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*, 2nd Edition (Cambridge University Press, Cambridge, 1992).

[7] J. P. Girardeau-Montaut and C. Girardeau-Montaut, "Space-charge-limited current density as a function of electron flow duration in an emissive diode," *Journal of Applied Physics*, vol. **65**, pp. 2889-2895 (1989).

[8] F. F Chen, *Introduction to Plasma Physics and Controlled Fusion*, 2nd Edition (Plenum, New York, 1984).

[9] A. R. Melnyk and J. R. Harrison, "Resonant Excitation of Plasmons in Thin Films by Electromagnetic Waves," *Physical Review Letters*, vol. **21**, pp. 85-88 (1968).

[10] S. E. Irvine and A. Y. Elezzabi, "Ponderomotive electron acceleration using surface plasmon waves excited with femtosecond laser pulses," *Applied Physics Letters*, vol. **86**, pp. 264102 (2005).

[11] S. E. Irvine, A. Dechant, and A. Y. Elezzabi, "Generation of 0.4-keV Femtosecond Electron Pulses using Impulsively Excited Surface Plasmons," *Physical Review Letters*, vol. **93**, pp. 184801 (2004).

[12] M. Aeschlimann, C. A. Schmuttenmaer, H. E. Elsayed-Ali, R. J. D. Miller, J. Cao, Y. Gao, and D. A. Mantell, "Observation of surface enhanced multiphoton photoemission from metal surfaces in the short pulse limit," *Journal of Chemical Physics*, vol. **102**, pp. 8606-8613 (1995).

[13] J. Kupersztych, P. Monchicourt, and M. Raynaud, "Ponderomotive Acceleration of Photoelectrons in Surface-Plasmon-Assisted Multiphoton Photoelectric Emission," *Physical Review Letters*, vol. **86**, pp. 5180-5183 (2001).

[14] C. Kan, Ph.D. thesis, University of Alberta, Edmonton, Alberta, 1997.

Chapter 4.0

Experimental Results

In this chapter, the details of the experiments used to study surface plasmon (SP) electron acceleration are described. Initial experimentation focused on implementing low intensity (sub GW/cm^2) pulses from a titanium-sapphire laser oscillator to generate energetic electrons from a silver metal film, and the energy spectra and photoemission process are measured. The electron emission dynamics are investigated using autocorrelation. Following this, the experiments are extended to higher intensities (multi GW/cm^2) using a laser amplifier system. Here, electron acceleration from both silver and gold surfaces is characterized according to kinetic energy, angular spectra, and two-pulse laser autocorrelation. Further measurements of the photocurrent variation with pump intensity are implemented to study the photoemission process from the silver and gold surfaces. Finally, the experimental results are compared with the model calculations described in Chapter 3 and conclusions are drawn.

A portion of this chapter has been published: S. E. Irvine and A. Y. Elezzabi, Physical Review Letters, **93**, 184801, 2004, Copyright (2004) American Physical Society; S. E. Irvine and A. Y. Elezzabi, Applied Physics Letters, **86**, 264102, 2005, Copyright (2005) American Institute of Physics.

4.1 Experiments Using Low-Energy High-Repetition Rate Femtosecond Pulses from a Laser Oscillator

One of the goals of this thesis was to implement a simple laser oscillator to study SP acceleration. To this end, the experimental arrangement shown in Figure 4.1 was designed and employed. The ultrafast laser system is a Kerr-lens mode-locked titanium-sapphire oscillator that is excited by a continuous-wave diode pumped Nd:YVO_4 laser

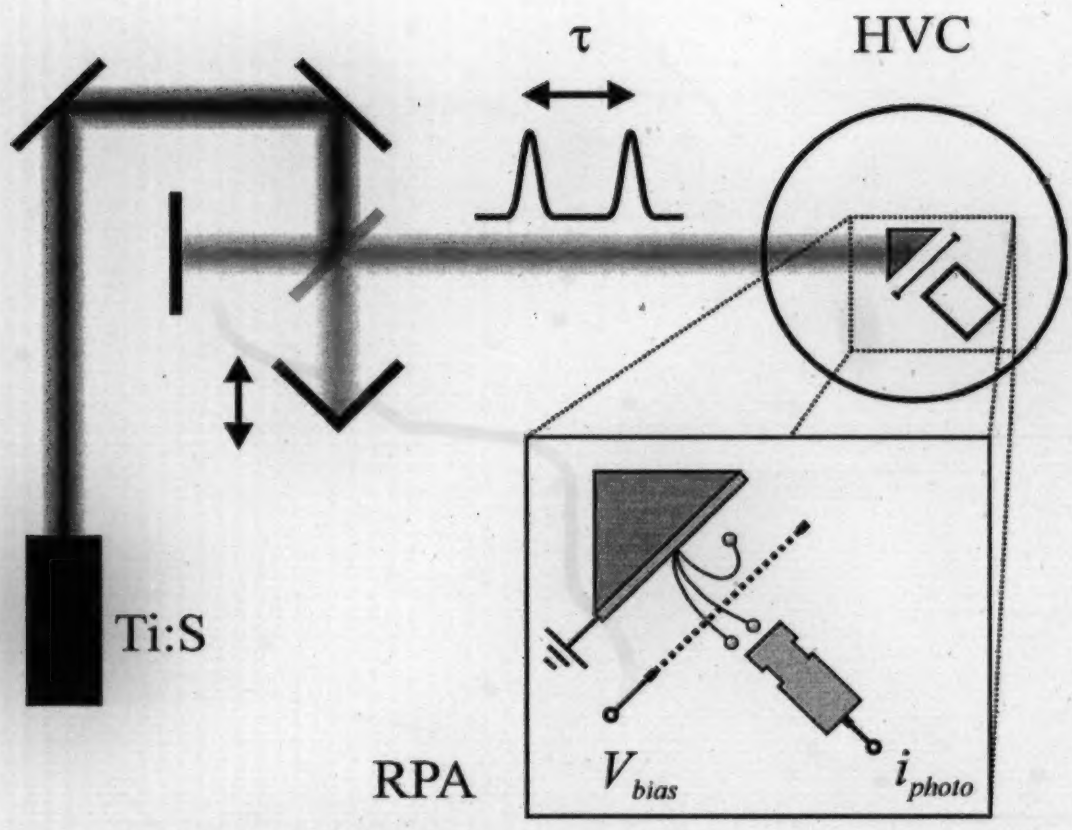

Figure 4.1. Experimental arrangement for SP electron acceleration. HVC: high vacuum chamber, Ti:S: Kerr-lens mode-locked titanium-sapphire laser oscillator, RPA: retarding potential analyzer. Energy spectra are determined by tracking the amplified photocurrent i_{photo} as a function of the grid voltage V_{bias}.

(Spectra Physics Millenia V) and is capable of delivering 8 fs, 1.5 nJ pulses at a repetition rate of 80 MHz. The transverse magnetic (TM) polarized optical pulses are direct from the output of laser system and through a Michelson interferometer arrangement, which provides two pulses having an equal intensity and a variable relative time delay, τ. The two delayed pulses are then directed to a vacuum chamber (evacuated to 10^{-5}-10^{-6} Torr) that contains the prism and the electron detection/characterization apparatus. Once the laser pulses enter the vacuum chamber through a silica window, the beam is focused (f =20 cm, 60 μm spot size) into the prism and impinges the d =~50 nm silver metal film that has been deposited via magnetron sputtering. The prism is fixed to a rotational mount that can be adjusted to minimize the optical beam that exits the prism thus maximizing coupling to SP waves.

To ascertain the electron photoemission mechanism, the dependence of the photocurrent on incident laser intensity is measured. Here, current generated from the surface of the prism was detected and monitored using either a copper Faraday cup and an electrometer (Keithley 619), or a single-channel electron multiplier (Ceremax 7596m, zero post acceleration, circuit in Figure A.3 of Appendix A) and a lock-in amplifier (Stanford SR830). The laser pulse intensity was selected by placing a variable neutral density filter in the optical beam path. Figure 4.2a displays the results of this experiment for intensities ranging from 0.03 to 0.95 GW/cm^2. At the highest power density, the average photocurrent was 1.3 nA, corresponding to 16 aC of charge per pulse. These results provide clear evidence that the electron emission is a three-photon photoemission process (m = 3), which is in good agreement with the Einstein photoemission equation

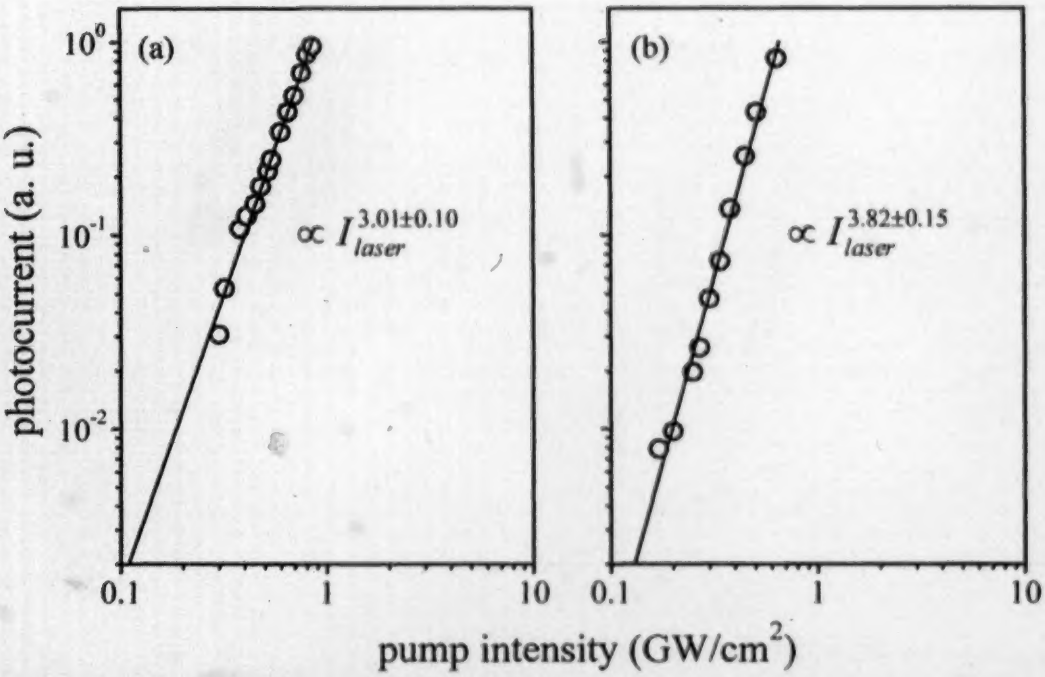

Figure 4.2. (a) Measured photocurrent dependence vs. intensity verifying a three-photon process for 800 nm laser oscillator pulses irradiating a silver metal film in the Kretschmann configuration. (b) Measured photocurrent dependence vs. intensity using 5 fs laser pulses from an ultra-broadband titanium-sapphire oscillator.

for a photon energy of 1.55 eV (at $\lambda_0 = 800$ nm) and silver work function $W_f = 4.3$ eV [1]. It is interesting to note that while the presence of the SP wave enhances the coupling of laser pulses into the film, the dominance of multiphoton photo-excitation supports the fact that the electron emission is correlated with the intensity of the laser pulse and not the electric field of the plasmon. No indications of tunnel effects are present, as the order of the emission process remains constant across all intensities of the experiment. The fact that the emission process is multiphoton in nature (over this intensity range) justifies a simple power scaling for electron emission in the model calculations.

Additional experiments confirming multiphoton electron emission were also performed at the Max-Planck Institute for Quantum Optics (Garching, Germany), which utilized the same experimental arrangement with the exception of the laser system. Instead, an ultra-broadband titanium-sapphire oscillator was employed, which was capable of generating 5 fs pulses at energies up to ~5 nJ. Consideration must be given to dispersion of the ultra-broadband 5 fs pulses within the prism. Hence, the optical pulses from the oscillator were pre-compressed using chirped mirrors and the pulse duration was optimized at the prism surface (in situ) by placing variable thickness silica plates within the beam path. Furthermore, coupling to SP waves was performed near the edge of the prism to minimize propagation through dispersive material. A similar photocurrent-intensity dependence was measured and is illustrated in Figure 4.2b, however, with a slightly higher order of emission of $m = 3.8$. The increase of the order can be attributed to two effects: (1) the significantly increased bandwidth of the laser source, which provides a larger range of photon energies and results in an effective mixed-order nonlinear dependence, and (2) the measurement was performed on a silver film that had been

exposed to atmosphere for a significant amount of time resulting in an over layer formation and an increase of the work function [2].

While the photocurrent-intensity traces provide striking evidence for a strictly multiphoton process, they offer no direct information of the emission dynamics at the surface of the metal film. Instead, higher-order autocorrelation using the delayed optical pulses is implemented and used as a measure of the coherence of the photoemission process. In the simplest case, the photocurrent generated at the prism surface can be represented by the m^{th} -order autocorrelation function:

$$i_{photo}(\tau) \propto \int_{-\infty}^{\infty} \left(\left[E_{laser}(t-\tau) + E_{layer}(t) \right]^2 \right)^m dt, \tag{4.1}$$

where m is the order of the process (equal to 3 for silver) and E_{laser} is the electric field:

$$E_{laser} = E_0 \operatorname{sech}\left(\frac{2t \operatorname{asech}\left(2^{-1/2}\right)}{\tau_p} \right) e^{i\left(\omega t + bt^2\right)} \tag{4.2}$$

where E_0 is the amplitude of the electric field, ω is the frequency, and b is the chirp parameter. When implementing a device for determining the pulse duration of an optical waveform, a suitable material having an instantaneous response (<few femtoseconds) is required (e.g. electro-optic crystals). If viewed in reverse, pulses of known duration can be used to probe the time-constant of an ultrafast optical process [3,4]. In the particular case here, the lifetime of the plasmon wave has been measured to be $\tau_{plasmon} = 48$ fs [5], and thus, if the electron emission truly resulted from an electric field driven process, an autocorrelation trace would exhibit substantial broadening corresponding to $\tau_{plasmon}$. Results of the autocorrelation experiment are shown in Figure 4.3 using pre-compensated (for material dispersion in glass) 5 fs laser pulses. The order of the correlation is 3.8 and

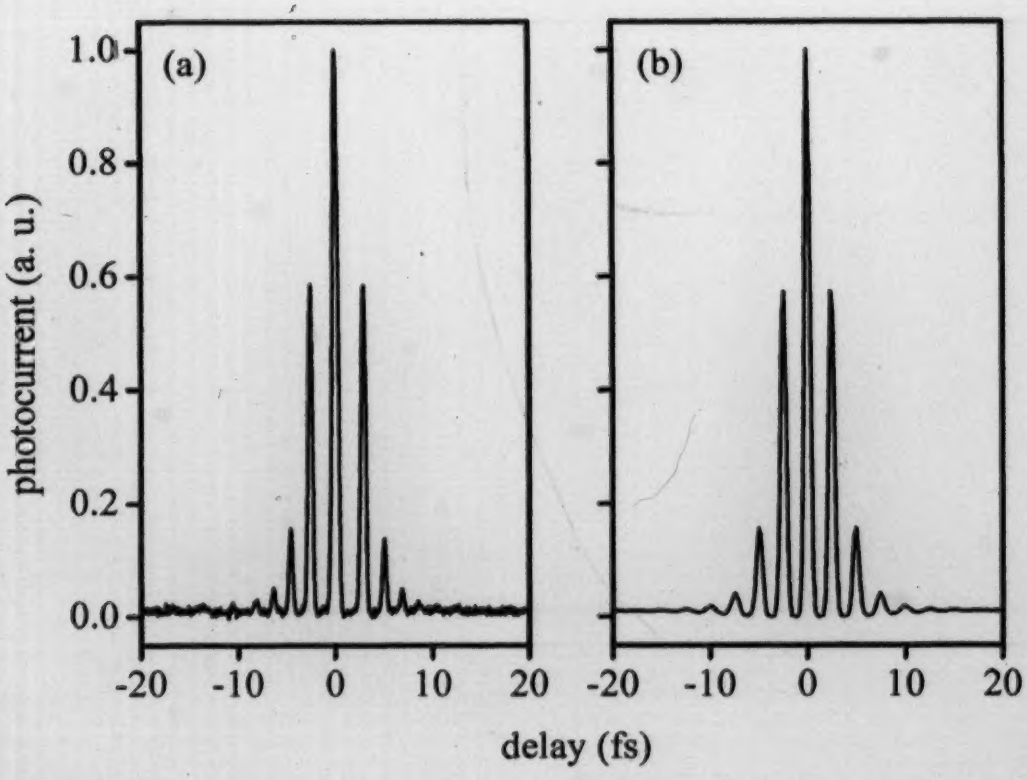

Figure 4.3. (a) Measured interferometric autocorrelation trace using 5 fs laser pulses from an ultra-broadband titanium-sapphire oscillator. (b) Calculated autocorrelation trace using equation 4.1 and $\tau_p = 5$ fs optical pulses. Comparison between (a) and (b) indicates that no broadening has occurred.

matches the value of m determined from the photocurrent-intensity characterization. More importantly, however, is the fact that no measurable broadening has occurred, as a comparison with the calculated autocorrelation function indicates that the optical pulses are 5 fs in duration. The measurements confirm that electron emission is correlated directly with the laser radiation and not with the electric field of the plasmon wave, and also provides additional motivation for using a simple power scaling for electron emission in the model calculations. The autocorrelation data in Figure 4.3 also negates the possibility that the increased order in Figure 4.2b is due to thermal enhancement effects, which take place on timescales of >1 ps [6,7].

To characterize the kinetic energy distribution of the photo-accelerated electrons, the retarding potential method is used. For this, the current emitted from the prism surface is monitored as a function of the voltage applied to a grid placed in front of the prism (see Figure 4.1). Due to the diminishing signal level at higher retarding voltages, the single-channel electron multiplier and lock-in detection scheme is preferred over the Faraday cup and electrometer detection method. It should be noted that the retarding potential technique of energy measurement provides the integrated spectra, as the grid acts as a high-pass filter in which electrons with energies beyond the bias voltage are detected. Hence, data acquired in this fashion must be differentiated to yield the true energy spectra.

Figure 4.4a illustrates a typical electrical signals obtained from the retarding potential analyzer. Each curve was obtained by optimizing the electron count at high retarding voltages (300-400 V) and represents the average of three distinct data sets. During the optimization procedure, it was observed that the electron signal was highly

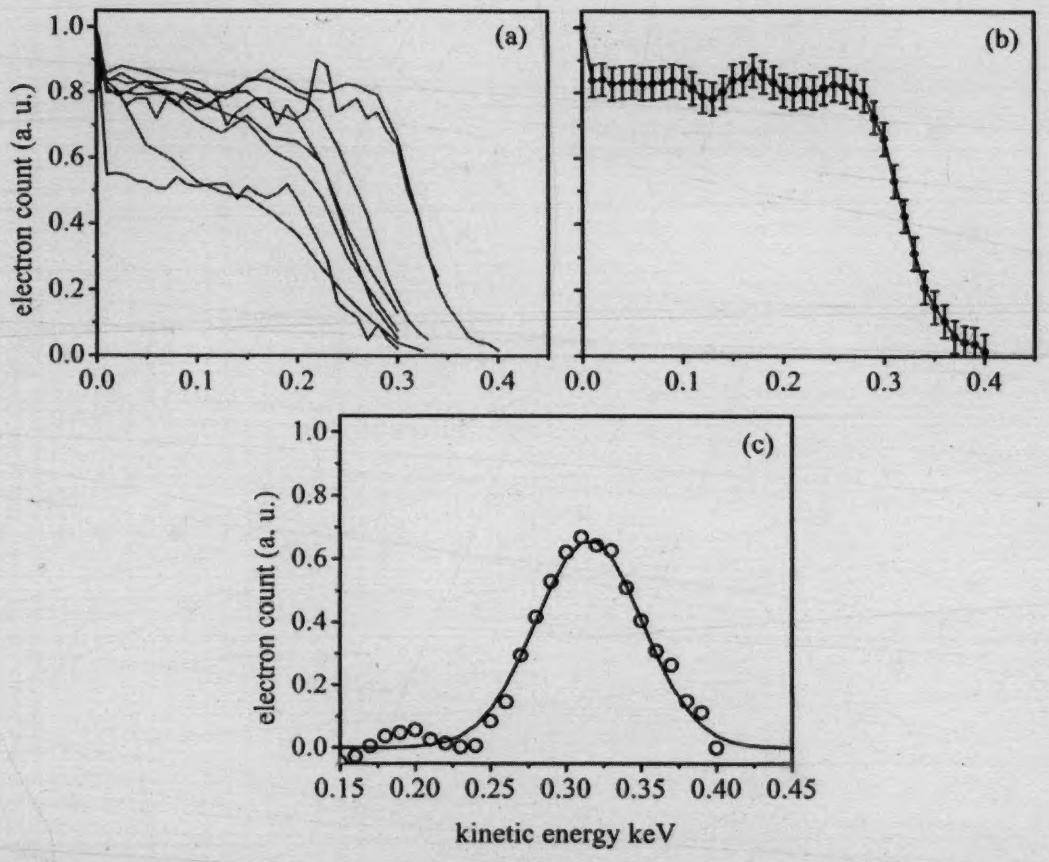

Figure 4.4. (a) Typical electrical signals obtained using the retarding potential analyzer using 1.5 nJ optical pulses from the titanium-sapphire laser oscillator at an intensity of 0.5 GW/cm^2. Each curve represents a different location on the prism surface. (b) Selected integrated spectrum illustrating that electrons with energies up to 0.4 keV are present. (c) Experimental kinetic energy distribution of the femtosecond electron pulses obtained by differentiating the data in panel (b).

sensitive to the location of the laser spot on the prism surface. This signifies that the surface of the silver metal film has a large degree of nonuniformity and plays an important role in the acceleration process and the shape of the resultant electron energy spectra. Figure 4.4b illustrates an integrated energy distribution that contains spectral components approaching 0.4 keV (also shown in Figure 4.4a). While a small number of low-energy electrons are present, the vast majority of electrons have energies above 0.25 keV. The low-energy cut-off, accounting for ~16% of the total number of electrons, is the contribution of thermally induced background emission (due to the high-repetition rate of the laser) and these electrons are not generated coherently with the laser radiation (as evidenced by the data of Figure 4.3). However, the high-energy electrons arise from the ponderomotive interaction at the surface of the metal film. Figure 4.4c shows the actual kinetic energy spectrum of the electron pulses obtained by differentiating Figure 4.4b. The most striking features of the high-energy distribution are its large central value of 0.315 keV, as well as its 83 eV full width at half maximum (only 26% of the central value). Given the fact that the acceleration takes place within the evanescent penetration depth of the SP wave (~240 nm), an effective acceleration gradient in excess of 1 GeV/m is calculated. Despite the fact that the intensity in this experiment ($I_{laser} = 0.5$ GW/cm^2) is more than four orders of magnitude lower than the previously reported values ($I_{laser} = 40$ TW/cm^2) produced from 150 fs pulses [8], similar maximum kinetic energies and much narrower spectra are obtained. Previous low-intensity experiments in gold by the same group [9] reported electron energies of only 40 eV at $I_{laser} = 21$ GW/cm^2. Indeed, the result illustrated here is a significant step towards the long-standing goal of generating energetic electrons via low power laser systems and opens the doorway to studying high-

field effects using simple titanium-sapphire oscillators. As well, all-optical electron acceleration at such a high repetition-rate would be extremely useful for improving the sensitivity of time-resolved experimentation based on electron probe pulses.

4.2 Experiments Using High-Energy Femtosecond Pulses from a Laser Amplifier

After demonstrating that electron acceleration could be achieved using low-energy pulses from a titanium-sapphire oscillator, the next step was to extend the experiments to higher intensities using a laser amplifier system. The test apparatus was similar to the arrangement shown in Figure 4.1, however, the distinct difference is the actual laser system involved. Instead of a simple oscillator, a multi-pass chirped-pulse laser amplifier was used. Within it is contained a dedicated titanium-sapphire laser oscillator (identical to that in the experiments described above) that provides seed pulses for a subsequent amplification stage. Pulses from the oscillator are reduced to a repetition rate of 1 kHz and directed through a titanium-sapphire crystal that is pumped by a Q-switched frequency-doubled Nd:YAG laser (Coherent Corona), which amplifies each pulse by a factor of $>10^5$. As a result, the amplifier system is capable of generating 30 fs pulses having 0.5 mJ of energy. The central wavelength and polarization of the amplified laser beam are $\lambda_0 = 800$ nm and transverse magnetic (TM), respectively.

The electron emission induced by the laser amplifier pulses is characterized according to SP coupling and directionality for a gold surface. Shown in Figure 4.5 is the reflectance of the optical beam in the Kretschmann configuration as a function of the angle of incidence, also known as attenuated total reflection spectra. Clear resonance behavior is observed for the gold metal film. The central resonance dip occurs at an angle

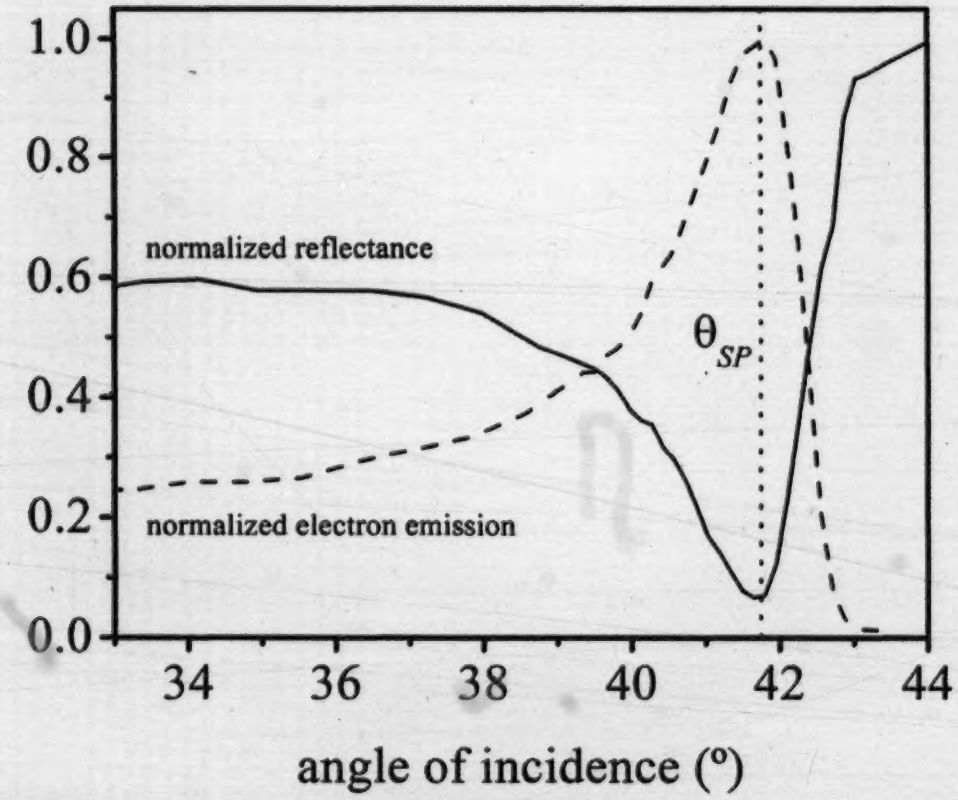

Figure 4.5. (solid line) Measured reflectance of the optical beam in the Kretschmann configuration using the laser amplifier system and a gold metal film. The resonant absorption dip clearly indicates coupling to SP modes. The resonance angle occurs at a value of θ_{SP} =41.7° (vertical dotted line) and the full-width at half-maximum is 4°. Also shown is the electron emission as a function of angle of incidence of the laser beam (dashed line), which indicates that photoemission of electrons is directly correlated with SP coupling.

of 41.7±0.1° and has a full-width at half-maximum of 4°. Also shown is the photocurrent dependence on angle of incidence, which also has a maximum (~5 nA corresponding to 5 pC per pulse) at an angle of 41.7±0.2°. It is observed that the photoemission effectively 'mirrors' the plasmon resonance curve, and thus, the photoemission of electrons is directly correlated with SP coupling. The directionality of the SP accelerated electron bunches is measured using the apparatus shown in Figure 4.6. A copper Faraday cup is swept about the prism surface from -90° to 90° at a radius of 5.5 cm and the collected photocurrent is measured using an electrometer. The measured angular distributions of electrons accelerated by SP waves on silver and gold surfaces are shown in the right panel of Figure 4.6. These distributions reveal that the electrons are highly directional with maximum current occurring at an angle of ~10°. Along this direction, the distributions are nearly symmetric having angular half widths of ~ 50°.

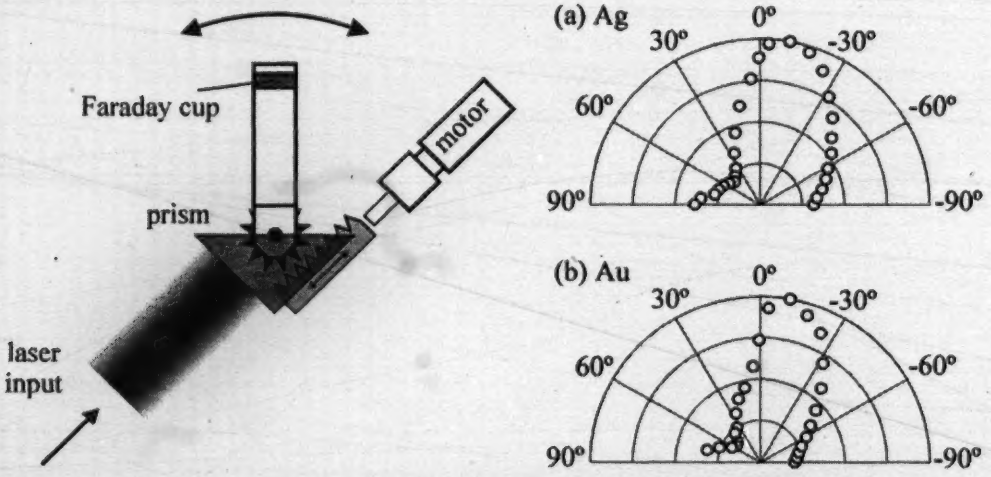

Figure 4.6. (left) Apparatus used to measure the angular distribution of the photo-accelerated electrons. A rack and pinion drive system is implemented to sweep a Faraday cup about the prism surface. The resulting spectra are plotted for both (a) silver and (b) gold. Peak emission occurs near 10° and the distribution are nearly symmetric about this direction with angular half-widths of ~50°.

The autocorrelation data described in the previous section accurately predicted the order of the multiphoton electron ejection, and therefore, is chosen as the starting point for examining the photoemission dynamics. Shown in Figure 4.7 are the measured autocorrelation traces for both silver and gold surfaces in which each pulse has an intensity of ~3 GW/cm^2. Also shown in Figure 4.7 are the theoretically calculated traces (from equation 4.1). The excellent agreement between theory and experiment indicates that the optical pulse has broadened to nearly 100 fs following its propagation through the prism and has acquired a significant amount of chirp ($b = 3.5\times10^{26}$ 1/s^2). Even more striking, however, is the fact that the order of the autocorrelations are 1.75 for silver and 1.65 for gold; much less than the expected 3rd and 4th orders for silver ($W_f = 4.3$ eV [1]) and gold ($W_f = 5.3$ eV [10]).

The reduced orders reveal that another photoemission process is taking place at these pump power densities. To investigate the emission process in greater detail, the photocurrent is monitored directly as a function of the power density as described in the previous section. The intensity irradiating the metal-coated prisms is varied from 0.5 to 12 GW/cm^2 and the corresponding photocurrent is recorded. In contrast to the previous observation using lower intensity laser oscillator pulses, the present photocurrent-intensity dependence, illustrated in Figure 4.8a and 4.8b, shows two distinct photoemission regimes for both silver and gold films. The first regime, extending up to power densities of $I_{laser} = 1.6$ GW/cm^2 for silver and $I_{laser} = 2.1$ GW/cm^2 for gold, is that of multiphoton photoemission. In this regime, 3rd and 4th order power dependencies are observed for silver and gold, respectively. However, for power densities above 3.1 GW/cm^2 irradiating the silver film, there is a transition over which m decreases to 1.47.

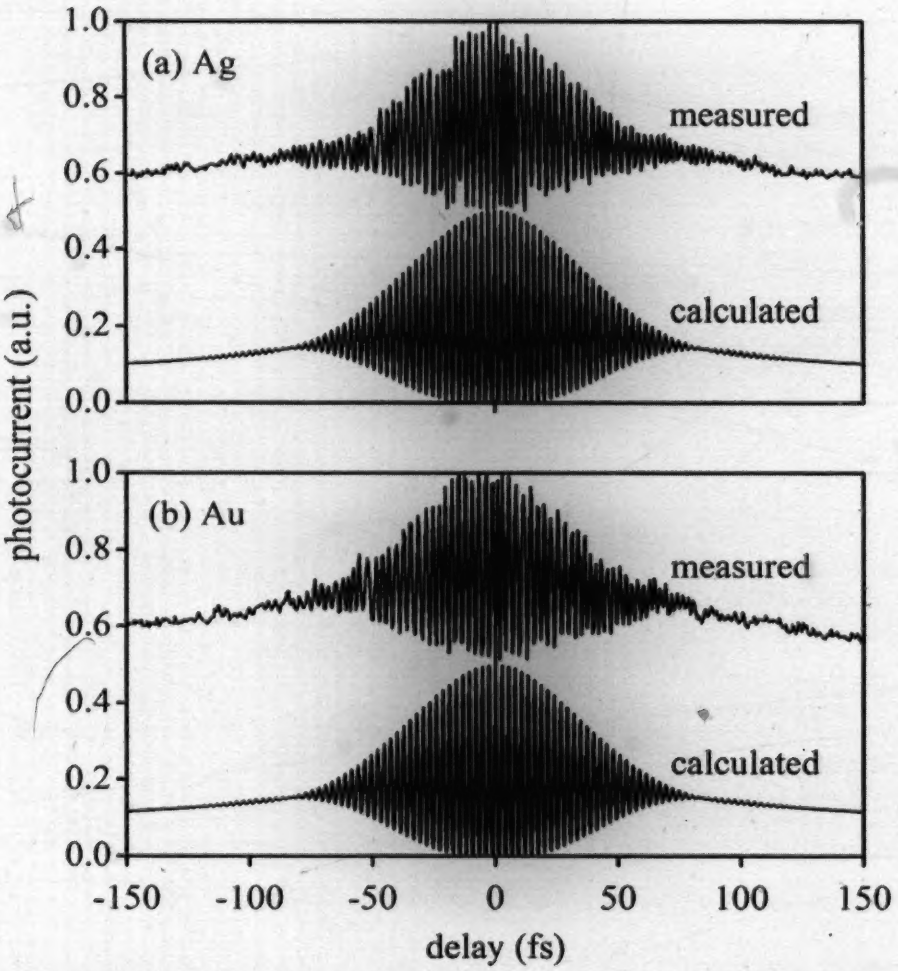

Figure 4.7. Calculated and measured interferometric autocorrelation traces for both (a) silver and (b) gold surfaces. The traces reveal that the optical pulse has broadened to 100 fs during its propagation through the prism. More striking is the fact that the orders of the autocorrelations are 1.75 for silver and 1.65 for gold; much less than the expected 3rd and 4th order dependencies.

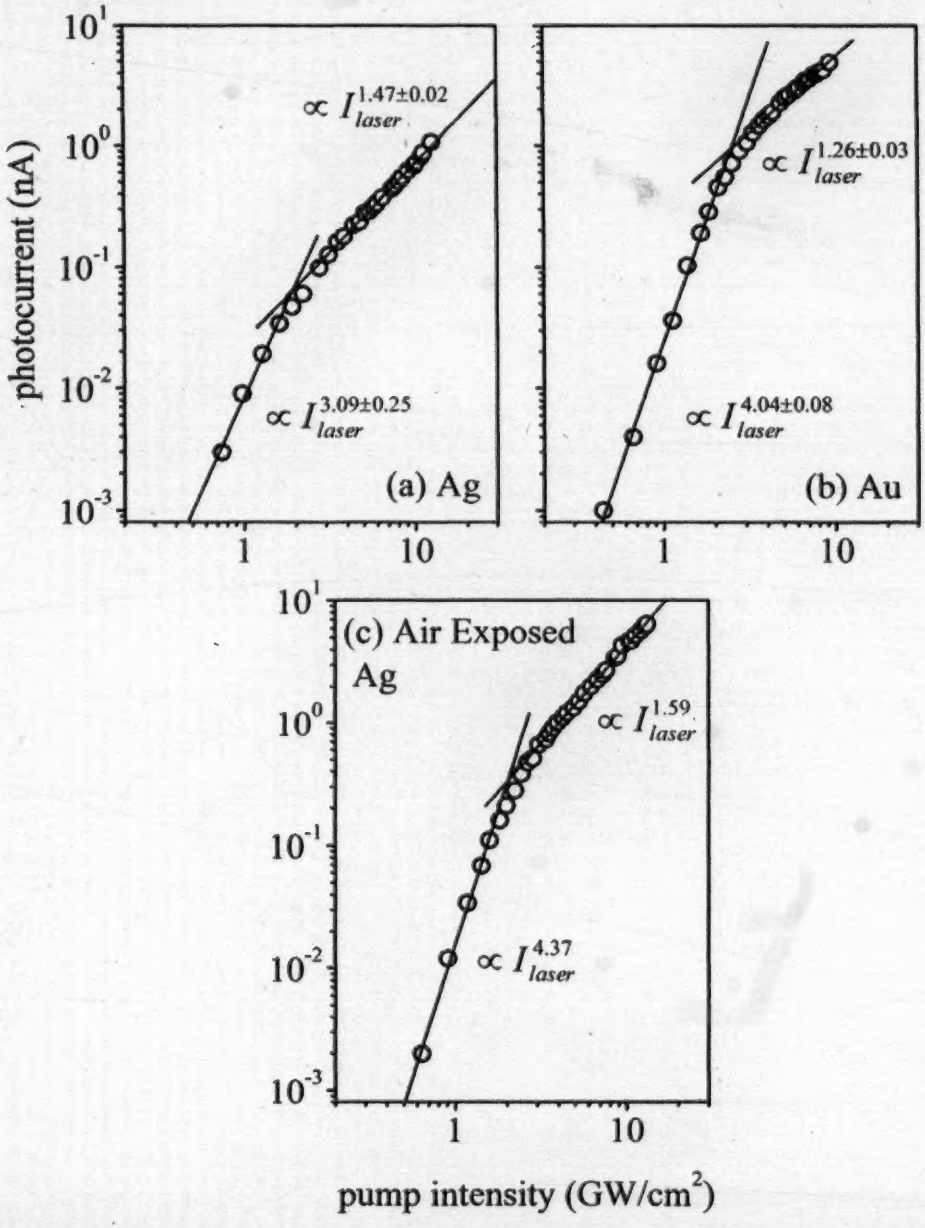

Figure 4.8. Dependence of photocurrent on pump intensity for (a) silver and (b) gold. For lower intensities, 3rd and 4th order power dependencies are observed. However, at higher intensities the orders are reduced in both metals to 1.47 and 1.26 for silver and gold, respectively. (c) A similar dependence of photocurrent on pump intensity is obtained for a silver film that has been exposed to atmosphere for ~24 hours. In this case the multiphoton order is increase, as is the order in the tunnel regime.

Similarly, for intensities above 3.3 GW/cm^2 in gold, the order is reduced to $m = 1.26$. In either case, there is an onset of another type of photoemission process. This photoemission cannot be thermionic for two reasons. First, in the case of thermally assisted multiphoton emission [11], the order is increased (not decreased) as the laser intensity is increased. Second, heat mediated electron emission takes place on a sub-picosecond timescale [12], which would result in broad (> 1 ps) shoulders in the autocorrelation data.

Both the silver and gold photocurrent-intensity traces show a distinct transition to another photoemission regime and resemble Keldysh ionization of atoms in intense laser-fields [13-15]. In the case of metallic films, the barrier height is given by the work function instead of the ionization potential. As the laser intensity is increased, the electric field produced at the surface reduces the potential barrier (work function) and causes electrons to tunnel directly into vacuum. Using the relevant laser parameters, the Keldysh parameter, γ, is calculated from equation 2.32 to be 134, indicating that multiphoton emission should dominate. Despite this fact, the evidence shown in Figures 4.8a and 4.8b indicates otherwise. Further results shown in Figure 4.8c also support the notion of electron tunneling from the metal film surfaces. The data presented here is the photocurrent-intensity trace of electron emission from a silver film that has been exposed to atmosphere for a period of approximately 24 hrs. Within this time span, significant tarnish (Ag_2S) layer formation has occurred and has allowed a dielectric layer (~0.4 nm [16]) to form over the silver film. As expected, the presence of the tarnish layer has both increased the multiphoton and tunnel regime orders as the presence of the oxide layer has effectively increased the work function.

Coupling of free-space radiation to SP modes results in an increase of the energy density of the wave (through confinement at the metal surface) that is manifested as an enhancement of the electric field. Thus, it is reasonable to conclude that it is this additional enhancement that allows access to the laser-induced tunnel regime. To arrive at an approximate value for the enhancement factor, the laser electric field and the electric field value required for tunnel effects are evaluated. Using the fact that $\gamma = 1$ marks the transition, an effective electric field of 1.6×10^{8} V/cm is calculated. When compared to the laser electric field value of 1.2×10^{6} V/cm at the transition intensity (~2 GW/cm^2), an electric field enhancement factor of $>10^2$ is deduced and is in good agreement with experimental values determined previously [8,9,17].

The enhancement of the electric field will also be evident in the velocity distribution of the electron bunches departing the prism surface. To explore the enhancement, the kinetic energy spectrum of the electron pulses is measured. While electron energy characterization can be carried out using the retarding potential method, alternative techniques are first explored due to their potentially decreased acquisition time and direct energy measurement capability (as opposed to providing the integrated spectra). One such method is shown in Figure 4.9a and is known as the time-of-flight technique. Here, the traversal time of the electrons across a known distance is used to quantify their velocities. The SP-coupling prism is directly mounted to the vacuum chamber, which consisted of a 0.38 m flight tube and a microchannel plate (MCP) array (Burle Technologies, Inc.), the latter of which serves as the electron detector. The solid angle subtended by the MCP array detector is 1.7×10^{-3} sr. Laser light enters the prism and generates an energetic electron pulse that travels the length of the tube and is

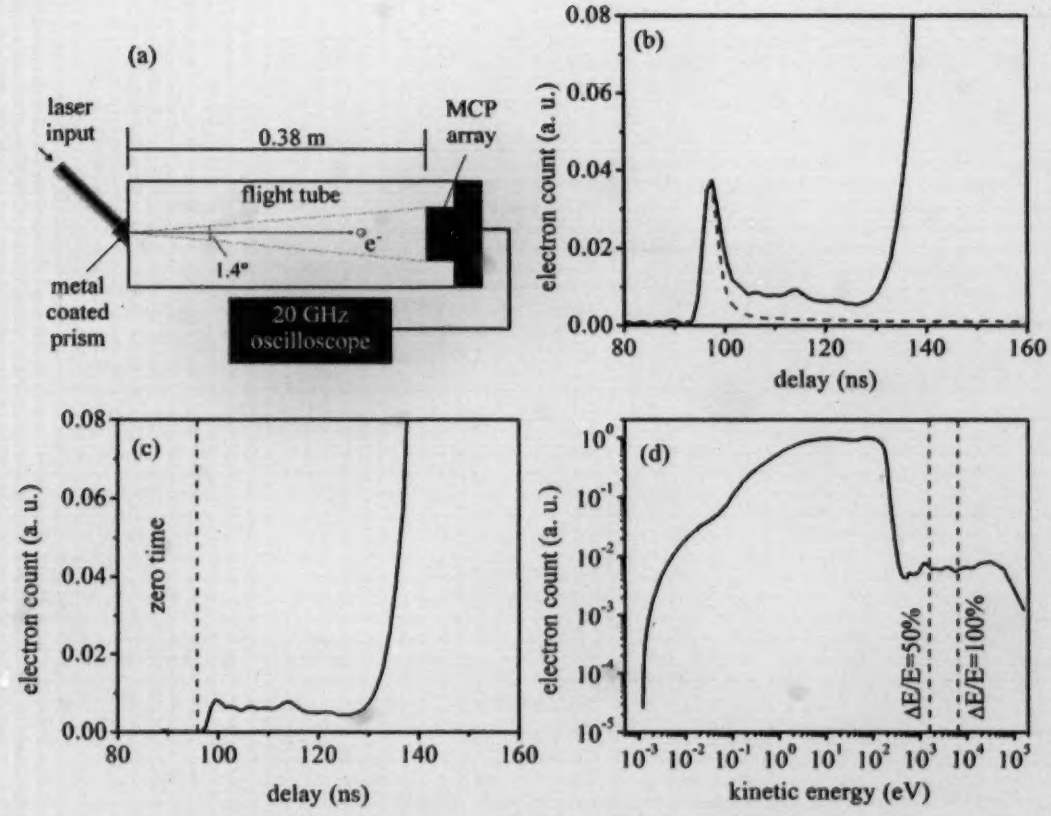

Figure 4.9. (a) Time-of-flight electron energy spectrometer. Electrons, emitted from the prism surface, travel the length of a 0.38 m flight tube and are detected by a microchannel plate (MCP) array. The signal from the MCP is processed using a 20 GHz oscilloscope (Tektronix CSA8000, 17.5 ps rise-time module). (b) The solid curve is the electrical waveform measured with the time-of-flight apparatus (close-up view of highest energy electrons). A dashed curve is also shown and corresponds to laser light that is sent directly through the apparatus without coupling to SP waves. (c) Signal produced from the difference of the waveforms shown in panel (b). The vertical lines in panels (b) and (c) indicate the zero time, which represents the time at which the electrons are emitted from the prism surface. (d) Kinetic energy spectrum of the photo-accelerated electrons determined from the waveform in (c) and a zero time of 95.9 ns.

subsequently detected by the MCP. Typically, a synchronization mechanism must be employed to accurately determine the time at which the electrons enter the flight tube. In this particular experimental apparatus, the MCP is capable of detecting the optical pulses, which are conveniently used to calibrate the spectrometer 'zero time'.

Figure 4.9b shows a typical electrical waveform measured with the time-of-flight setup at a laser intensity of 17 GW/cm^2 irradiating a gold-coated prism. The Gaussian peak located at 97.2 ns corresponds to light that has scattered from the prism surface and traveled the length of the tube. To remove this peak from the actual spectra, the prism is replaced with a window and the laser pulses are allowed to propagate directly to the detector. The resulting waveform is stored and subtracted from the combined optical-electron signal to yield a waveform that is due strictly to electrons and the result is plotted in Figure 4.9c. Also shown is the zero time, which is calculated to occur 1.3 ns ($0.38/c$) before the Gaussian peak located at 97.2 ns. Clearly, electrons can be detected almost immediately after the optical pulse. The corresponding kinetic energy spectrum is shown in Figure 4.9d and illustrates that electrons with a broad range of energies are present. A maximum occurs at a value of ~90 eV and a large cutoff in the electron count occurs at 250 eV. Notably, the spectral components are present beyond energy values of 1.6 keV. However, the uncertainty in measurement at this energy is 50% (determined from the 4 ns electronic settling time of the MCP). In order to reach a sufficient spectrometer resolution (~10%) at energies near 1.6 keV, the length of the flight tube would have to be increased to 2 m, and thus, the corresponding signal level would suffer a concomitant decrease by a factor of 25.

As an alternative to the time-of-flight technique, a novel magnetic sector electron spectrometer is designed and implemented. A detailed description of its construction and characterization are found in Appendix A of this thesis. Briefly, it is comprised of two coils and a central vacuum chamber through which electrons flow. The coils generate a static magnetic field and bend an electron beam through a fixed radius of curvature, r_0. Through knowledge of the magnetic field distribution at a given current in the coils, I, the energy of an electron source, which in this case is a gold-coated prism, can be measured directly by varying I and tracking the corresponding current reaching the detector (see the inset of Figure 4.10). By placing apertures along the electron path in the spectrometer, a resolution of 10% can be achieved for energies ranging up to 50 keV.

The resulting electron energy spectrum measured using the magnetic sector analyzer is shown in Figure 4.10 for a laser intensity of 36 GW/cm^2. Again, a large range of spectral components are present with a peak at 0.15 keV having a full-width at half-maximum of 0.19 keV. The broad range of energies results from the fact that the electrons sample many peak intensity and phase values during their photoemission into the SP wave. Due to the large traversal path (6 cm) of the electron packet, the signal is significantly reduced before reaching the detector and the resulting maximum detected electron energy is 1.1 keV. The estimated solid angle of detection is 7.8×10^{-3} sr and therefore, it is expected that if the signal level can be further improved by increasing the number of detected electrons, the maximum detected electron energy should also increase.

To achieve increased sensitivity, the retarding potential analyzer is implemented. The spectrometer is shown in Figure 4.1 and has a resolution of ~10% [18]. More

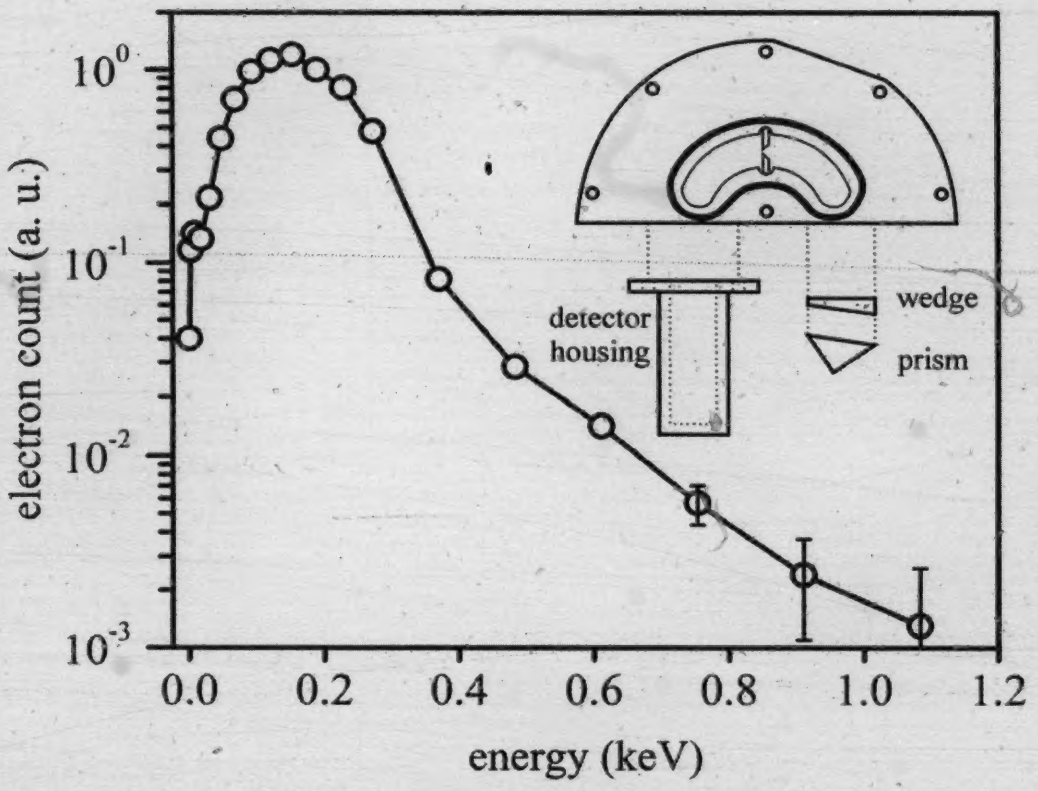

Figure 4.10. Kinetic energy spectrum measured using the magnetic sector analyzer described in Appendix A. Error bars show the reduced sensitivity at lower signal levels. The inset illustrates the configuration of the device, which included a 10° wedge to improve the signal yield (see Figure 4.6). To achieve a device resolution of 10%, two apertures (not shown) are placed at the electron source (gold-coated prism) and detector and have widths of 2 mm and 1 mm, respectively. An additional 1 mm aperture is placed mid-way through the central chamber arc.

important is its ability to perform the energy distribution measurements in close proximity (~1 cm) to the prism surface, which augments the solid angle of collection to 0.28 sr. Figure 4.11 illustrates the integrated kinetic energy distributions of the electrons measured using the retarding potential method. Electrons with energies ranging up to 2 keV are generated from both silver and gold films, which given the characteristic SP penetration depth of 240 nm, yields an acceleration gradient of ~8 GeV/m. The irradiating intensity was ~10 GW/cm^2 for the silver and gold surfaces, corresponding to an electric field strength of 2.8×10^6 V/cm. Irrespective of the metals used, both curves exhibit a similar monotonic decrease with half-width at half-maxima of 0.66 kV and 0.65 kV for silver and gold, respectively. The integrated kinetic energy distributions reveal that 34% of the photoelectrons from silver and 24% from gold have energies in excess of 1 keV.

4.3 Comparison of Model Calculations with Experimental Measurements

Shown in Figure 4.12 is a measured electron energy spectrum (originally shown in Figure 4.4c) obtained using laser oscillator pulses. The irradiating intensity was 0.5 GW/cm^2, which corresponds to an electric field strength of approximately 10^6 V/cm. Also illustrated is a calculated energy distribution (via the model in Chapter 3) using an SP electric field of $E_{SP} = 1.8 \times 10^9$ V/cm. The excellent agreement between the experimental and theoretical spectra indicates that the electric field is enhanced by over three orders of magnitude; much higher than the predicted enhancement of $\eta = 3.7$ discussed in Chapter 3. Depending on the exact nanometric surface morphology and the proximity of the adjacent defects, electric field enhancement factors above 250 can be achieved [19,20].

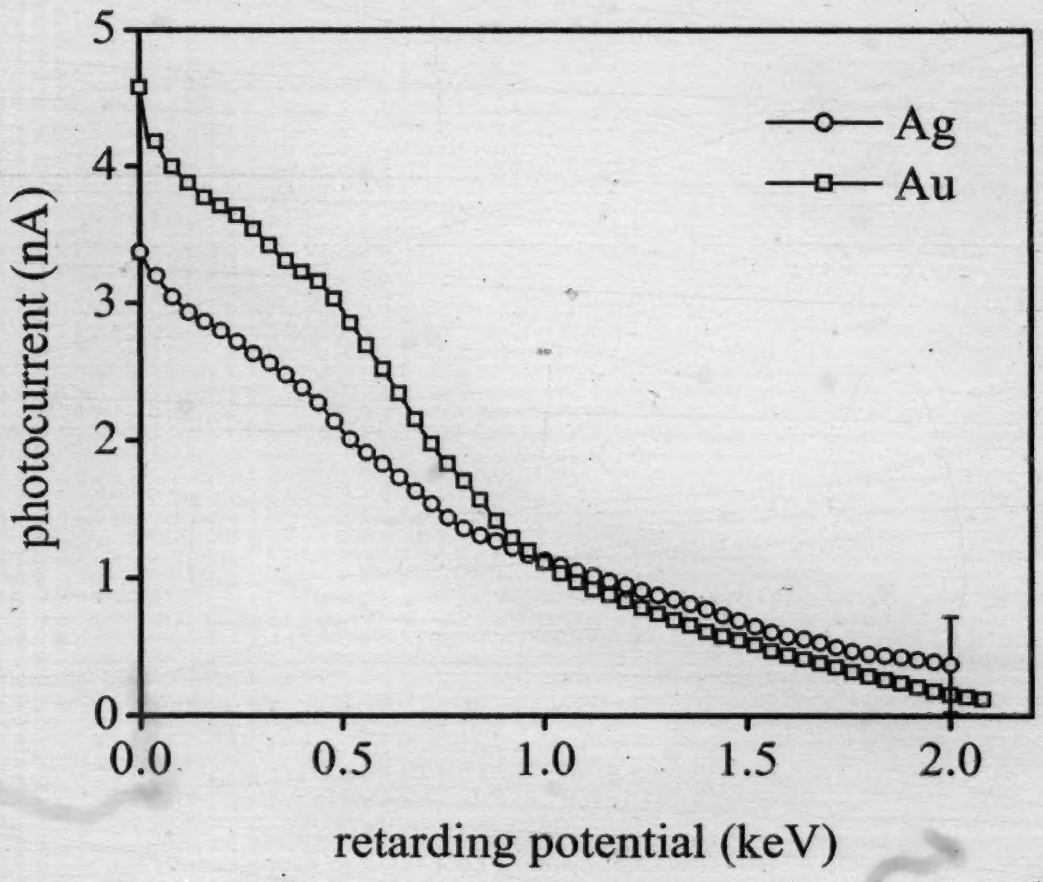

Figure 4.11. Integrated kinetic energy spectrum measured using the retarding potential method shown in Figure 4.1. A broad range of spectral components are present from 0 to 2 keV for both silver (circle) and gold (square) surfaces. The error is indicated only on the final point of the silver surface curve for clarity and applies to all data points.

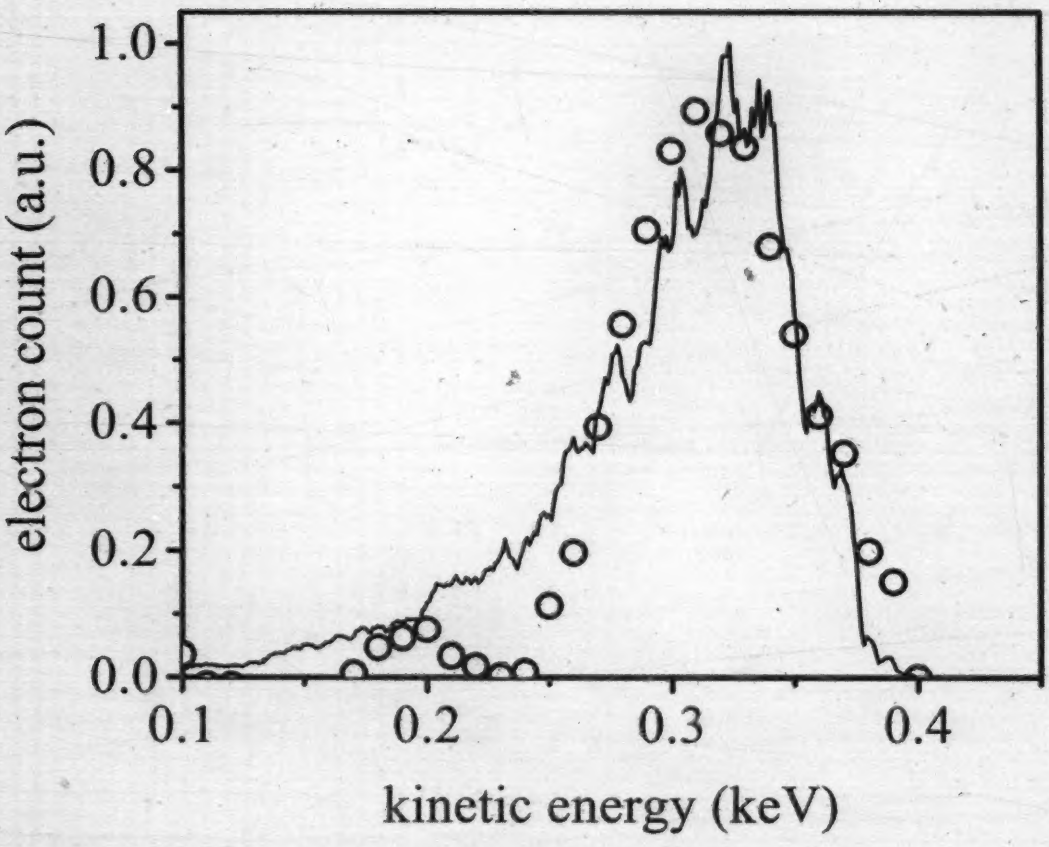

Figure 4.12. Comparison between measured electron energy spectra using the laser oscillator (circles) and theoretical energy spectra (solid) as calculated from the model described in Chapter 3.

Furthermore, such a narrow theoretical spectrum can only be obtained if the electron emission is limited to within 10% of the laser spatial full-width at half-maximum. The increased enhancement and confined electron emission can be explained by considering the surface morphology of the silver film. Surface roughness effects alter the spatial distribution of the SP field on a nanometer scale (< 50 nm) and are not included in the model calculations. In such cases, the overall energy of the pulse is conserved, but the energy density is drastically increased by confinement of the radiation to sub-wavelength volumes and is manifested as an additional localized electric field enhancement. This explanation is further supported by the fact that the modeled electron emission must be restricted to within 10% of the laser spot. Due to the highly nonlinear photoemission, small peaks or protrusions at the metal surface would dominate the electron emission in the presence of an SP wave, and it would appear that electrons originate only from such defects having a reduced spatial extent. A full account of surface roughness necessitates three-dimensional FDTD calculation, which over the length scales of electron emission and acceleration, requires enormous computational effort in comparison to the current model and is not feasible at this time. Solution of this problem is realized by considering an overall effective electric field enhancement factor, η, which is determined by comparing the model results with those of the experiment, as shown in Figure 4.12. Since the two are in excellent agreement, an effective enhancement factor of $\eta \sim 10^3$ is ascertained.

Further evidence for surface roughness enhancement is present in the energy spectra obtained using the laser amplifier system. Figure 4.13 shows experimental spectra generated using amplified optical pulses having peak field strengths of 2.8×10^6 V/cm.

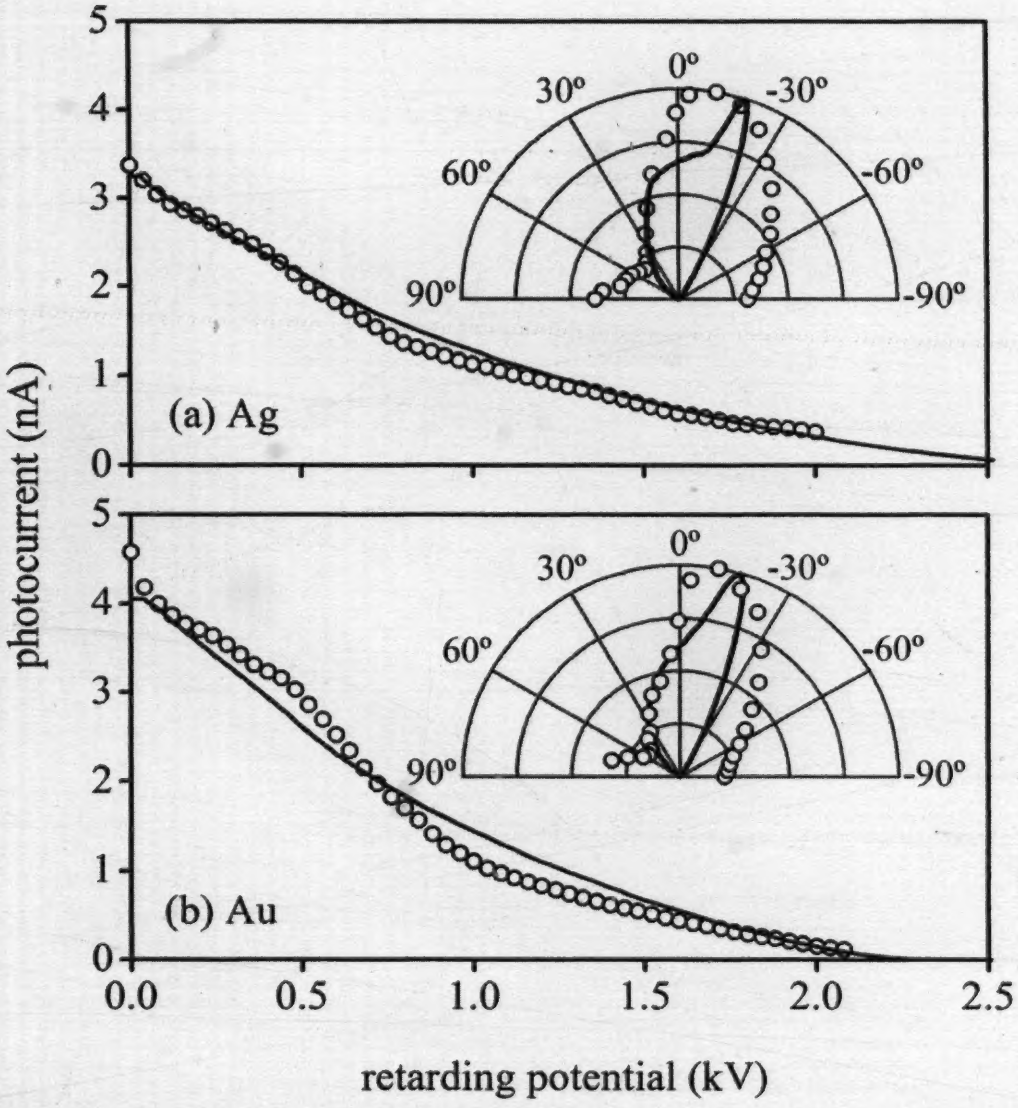

Figure 4.13. Calculated (solid lines) and experimentally measured (circles) integral kinetic energy distributions for (a) silver and (b) gold metal films. The calculated (solid lines) and experimentally measured (circles) in-plane angular distributions are shown in the corresponding insets.

Also shown are the corresponding calculated energy spectra using an SP field strength of $E_{SP} = 2.6 \times 10^9$ V/cm. In this case, electron emission was not restricted in the model calculations, which produced a much broader range of electron energies. Furthermore, the modeled functional dependence of photoemission is now given by the curves shown in Figure 4.8 instead of using a simple power scaling (as was the case for the laser oscillator spectra). Excellent agreement between the experimental and theoretical energy distributions is obtained and verifies an electric field enhancement of $\eta \sim 10^3$. Equally important is the obvious difference between the kinetic energy spectra produced using laser oscillator and the distributions generated using laser amplifier shown in Figure 4.13: the energy spectra produced by the oscillator are much narrower.

The discrepancy between the energy distributions obtained using the laser oscillator and laser amplifier systems is inherent to the experimental conditions and provides yet another clue for surface roughness. The laser oscillator generated electron energy spectra were produced using a laser beam that was focused to a spot size of ~60 μm and only the rays near the optical axis were efficiently coupled to the SP oscillation. This is verified by data obtained using a focused helium-neon laser beam, which is illustrated in Figure 4.14a. The thin dark vertical section in the center of the reflected beam indicates that only 6% is actually coupled to SP waves. In terms of the experiments employing the laser oscillator, this provided a physical mechanism for reducing the electron emission area. As such, electron emission was extremely sensitive to surface irregularities and resulted in surface 'hot spots', also consistent with metal photoemission experiments by another independent research group [21]. The previously discussed simulated laser oscillator energy spectra, in which electrons were confined to within 10%

of the spatial full-width at half-maximum of the optical pulse, indicated that the electron emission was highly localized and supports the concept of 'hot spots' resulting from surface roughness. Moreover, experiments involving the laser amplifier utilized a large (~8 mm) amplified beam that was collimated, thereby 1) allowing efficient coupling of the entire beam to the SP mode (also illustrated in Figure 4.14a) and 2) averaging over many surface irregularities. In terms of the model calculation, these two effects correlate to placing test electrons along the entire amplifier beam and the excellent agreement between the laser amplifier spectra and model calculations further supports this explanation. Also shown in Figure 4.13 are the calculated and experimentally measured angular distributions. Effective emission angles of 16° and 18° are calculated for silver and gold, respectively. The differences in the angular spectra are also attributed to inhomogeneities on the metal surfaces. In the vicinity of a local surface defect, the electric field will be highly distorted and will cause electrons to be emitted from the metal surface at various angles. This is consistent with the experimental angular distributions, which indicate that electrons are presents at all angles from -90° to 90°. Despite these facts, however, both spectra exhibit similar tilting of maximum emission away from the surface normal.

Direct confirmation of the hypothesized surface roughness is provided in Figure 4.14. An atomic force microscope image (Nanosensors PointProbe Plus, 7 nm tip radius) of a silver film deposited via magnetron sputtering (Nanofabrication Facility, University of Alberta) is shown in Figure 4.14b and clearly illustrates the inhomogeneous topology of the metal surface. Localized defects present on the film surface have heights up to 50 nm, a mean width of ~150 nm, and are separated by an average distance of 2 μm.

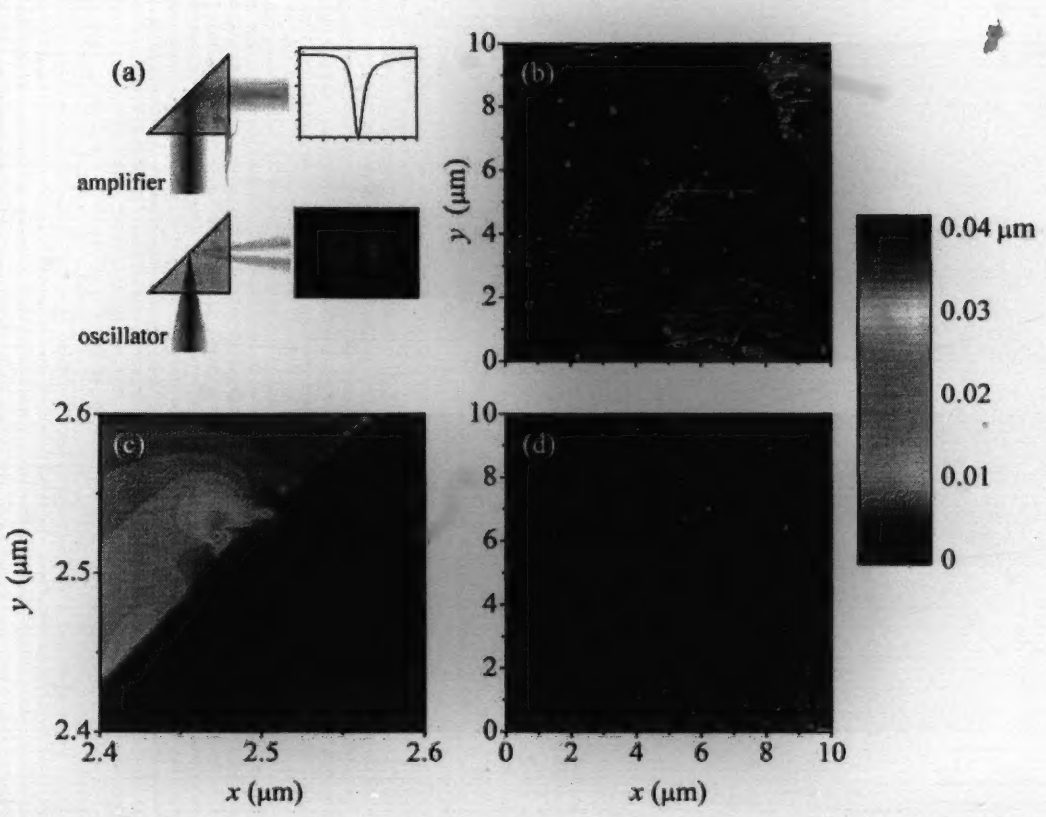

Figure 4.14. (a) Graphic illustration comparing the focusing conditions for the laser amplifier and laser oscillator experiments. For the amplifier, the entire beam is coupled to SP waves as each portion of the beam has the same propagation vector. In contrast, the oscillator beam is focused, and therefore, only rays near the optical axis have the necessary momentum to couple to SPs. The image next the depiction of the oscillator illustrates a measurement taken using a focused helium-neon laser beam. It is clear that only the central portion (6%) is efficiently coupled. Panels (b) and (d) show atomic force microscope images of two silver surfaces deposited by (b) magnetron and (d) ion-beam sputtering techniques. The vertical scale (color scale) on the right indicates that the surface features of (b) have heights up to 40 nm. Graph (c) is a two-dimensional FDTD simulation of the electric field distribution in the vicinity of a surface defect (25 nm pyramid). The additional enhancement due to confinement is clear ($\eta \sim 12$).

Considering the small spot size (60 μm diameter) and coupling conditions of the laser oscillator beam (6%), it is clear that only one or two such 'hot spots' would be excited by the laser radiation and lead to confined electron emission. A two-dimension FDTD simulation of the SP field distribution in the vicinity of a ~25 nm pyramid is shown in Figure 4.14c and indicates additional enhancement ($\eta \sim 12$). While the two-dimensional code is unable to accurately calculate the true enhancement for rough surfaces (i.e. three-dimensional topologies), it clearly illustrates the electromagnetic confinement near the apex of the pyramid. A similar atomic force microscope image of a silver surface that has been deposited using ion-beam sputtering (Department of Physics, Ludwig-Maximilians University, Garching, Germany) is shown in Figure 4.14d and is void of any surface defects resembling those in Figure 4.14b. When implemented for SP electron acceleration, zero electrons are present with energies beyond ~5 eV.

4.4 Summary

The results of this chapter indicate that SP waves are an effective means of generating energetic electrons. Initial experiments focused on the excitation of SP waves using low-energy pulses from laser oscillator. Here, it was shown that electrons with energies ranging up to 0.4 keV could be generated using only 1.5 nJ pulses ($I_{laser} = 0.5$ GW/cm^2). The finding marks an important and timely accomplishment, as it opens the possibility of studying high-field effects using a high-repetition rate low-energy (and relatively inexpensive) laser oscillator system. Further experiments using higher energy pulses (0.5 mJ) from a laser amplifier indicated, for the first time, that electrons with energies ranging up to 2 keV could be generated at intensities of 10 GW/cm^2. A detailed

examination of the photo-ejection process reveals that electron emission characteristics depart from multiphoton absorption towards Keldysh-like field ionization for power densities above ~2 GW/cm^2. The experimental kinetic energy spectra and angular distributions are also in agreement with the modeled results. Based on the agreement of the energy spectra, it is determined that the electric field is enhanced by over 10^3. Through comparison of the experimental and calculated spectra, it is determined that surface roughness of the metallic films plays an important role in the enhancement of the electric field.

4.5 References

[1] M. Chelvayohan and C. H. B. Mee, "Work function measurements on (110), (100) and (111) surfaces of silver," *Journal of Physics C: Solid State Physics*, vol. **15**, pp. 2305-2312 (1982).

[2] M. Akbi and A. Lefort, "Work function measurements of contact materials for industrial use," *Journal Of Physics D*, vol. **31**, pp. 1301-1308 (1998).

[3] H. Petek and S. Ogawa, "Femtosecond time-resolved two-photon photoemission studies of electron dynamics in metals," *Progress in Surface Science*, vol. **56**, 239-310 (1997).

[4] B. Lamprecht, J. R. Krenn, A. Leitner, and F. R. Aussenegg, "Resonant and Off-Resonant Light-Driven Plasmons in Metal Nanoparticles Studied by Femtosecond-Resolution Third-Harmonic Generation," *Physical Review Letters*, vol. **83**, pp. 4421-4424 (1999).

[5] M. van Exter and A. Lagendijk, "Ultrashort Surface-Plasmon and Phonon Dynamics," *Physical Review Letters*, vol. **60**, pp. 49-52 (1988).

[6] R. Y., J. Liu, and N. Bloembergen "Thermally assisted multiphoton photoelectric emission from tungsten," Optics Communications, vol. **35**, pp 277-282 (1980).

[7] H. E. Elsayed-Ali, T. B. Norris, M. A. Pessot, and G. A. Mourou, "Time-Resolved Observation of Electron-Phonon Relaxation in Copper," *Physical Review Letters*, vol. **58**, pp. 1212-1215 (1987).

[8] J. Zawadzka, D. Jaroszynski, J. J. Carey, and K. Wynne, "Evanescent-wave acceleration of ultrashort electron pulses," *Applied Physics Letters*, vol. **79**, pp. 2130-2132 (2001).

[9] J. Zawadzka, D. Jaroszynski, J. J. Carey, and K. Wynne, "Evanescent-wave acceleration of femtosecond electron bunches," *Nuclear Instruments and Methods in Physics Research A*, vol. **445**, pp. 324-328 (2000).

[10] W. M. H. Sachtler, G. J. H. Dorgelo and A. A. Holscher, "The work function of gold," *Surface Science*, vol. **5**, pp. 221-229 (1966).

[11] R. Yen, J. Liu, and N. Bloembergen, "Thermally Assisted Multi-Photon Photoelectric Emission from Tungsten," *Optics Communications*, vol. **35**, pp. 277-282 (1980).

[12] R. W. Schoenlein, W. Z. Lin, J. G. Fujimoto, and G. L. Eesley "Femtosecond Studies of Nonequilibrium Electronic Processes in Metals," *Physical Review Letters*, vol. **58**, pp. 1680-1683 (1987).

[13] C. Tóth, G. Farkas, and K. L. Vodopyanov, "Laser-Induced Electron Emission from an Au Surface Irradiated by Single Picosecond Pulses at λ=2.94 μm. The

Intermediate Region Between Multiphoton and Tunneling Effects," *Applied Physics B*, vol. **53**, pp. 221-225 (1991).

[14] C. Guo, "Multielectron Effects on Single-Electron Strong Field Ionization," *Physical Review Letters*, vol. **85**, pp. 2276-2279 (2000).

[15] V. S. Popov, "Tunnel and multiphoton ionization of atoms and ions in a strong laser field (Keldysh theory)," *Physics-Uspekhi*, vol. **47**, pp. 855-885 (2004).

[16] H. E. Bennett, R. L. Peck, D. K. Burge, and J. M. Bennett, "Formation and Growth of Tarnish on Evaporated Silver Films," *Journal of Applied Physics*, vol. **40**, pp. 3351-3360 (1969).

[17] J. Kupersztych, P. Monchicourt, and M. Raynaud, "Ponderomotive Acceleration of Photoelectrons in Surface-Plasmon-Assisted Multiphoton Photoelectric Emission," *Physical Review Letters*, vol. **86**, pp. 5180-5183 (2001).

[18] C. L. Enloe and J. R. Shell, "Optimizing the energy resolution of planar retarding potential analyzers," *Review of Scientific Instruments*, vol. **63**, pp. 1788-1791 (1992).

[19] E. Hao and G. C. Schatz, "Electromagnetic fields around silver nanoparticles and dimers," *Journal of Chemical Physics*, vol. **120**, pp. 357-366 (2004).

[20] M. Stockman, private communication.

[21] M. Aeschlimann, C. A. Schmuttenmaer, H. E. Elsayed-Ali, R. J. D. Miller, J. Cao, Y. Gao, and D. A. Mantell, "Observation of surface enhanced multiphoton photoemission from metal surfaces in the short pulse limit," *Journal of Chemical Physics*, vol. **102**, pp. 8606-8613 (1995).

Chapter 5.0

Further Model Predictions

This chapter describes two novel physical effects related to SP electron acceleration that are analyzed using the model described in Chapter 3. First, an all-optical technique for femtosecond gating of electron beams/pulses using SP waves is proposed. Here, SP waves are used to gate an external electron beam that is directed toward the metal film surface from an arbitrary source. Since the SP field is produced on an ultrafast timescale, a significant fraction of the incoming electron beam can be sliced to yield electron packets having durations comparable to that of the excitation optical pulse. This technique hold great promise for the generation and characterization of ultrashort electron bunches on timescales below 100 fs, which currently remains an unsurpassed barrier [1].

The second effect is the influence of the carrier-envelope phase (CEP) of the optical waveform on SP electron acceleration. As shown below, the ponderomotive energy gain experienced by an electron in the electric field of an SP wave can be controlled through the CEP. When the SP wave is excited with a few-cycle laser pulse, spectral shifts within the electron energy distributions are observed and are correlated with the specific form of driving electric field oscillation of the light wave. Thus, a method for coherent optical manipulation of the acceleration process is afforded through the CEP of the light field. This is extremely important for the development of CEP calibration devices for low-energy few-cycle laser systems and is a step towards optical control of ultrafast solid-state processes.

A portion of this chapter has been published: S. E. Irvine and A. Y. Elezzabi, Optics Express, **14**, 4115-4127, 2006, Copyright (2006) Optical Society of America, Inc.; S. E. Irvine, P. Dombi, Gy. Farkas, and A. Y. Elezzabi, Physical Review Letters, **97**, 146801, 2006, Copyright (2006) American Physical Society.

5.1 Electron Beam Slicing and Gating

5.1.1 Geometry for Electron Beam Slicing

The underlying arrangement for slicing electron beams/pulses is based on the launching SP waves and is illustrated in Figure 5.1. An ultrashort optical pulse, of duration τ_p, enters a prism and strikes a thin metal film that has been deposited on the prism's surface. To generate ultrashort electron pulses, electrons are injected in the SP field. In contrast to the electron acceleration geometries discussed in the previous chapters, a continuous stream of electrons is externally directed towards the metal film and enters the SP field at an angle of incidence, θ, as shown in Figure 5.1. Once the electrons interact with the SP wave, they will experience the effective time-average ponderomotive force, which in this arrangement, is approximately normal to the metal film surface. If the kinetic energy of an incident electron is less than the ponderomotive potential, U_p, created by the SP field, then the electron will be deflected and depart the surface at angle, α, also shown in Figure 5.1. An electron beam, however, is comprised of many electrons having various arrival times and locations with respect to the peak of the SP field. In the region of spatial overlap between the electron beam and the SP field, only a finite portion of the electron beam will experience a change in momentum and will be redirected away form the prism's surface. The spatial extent of the sliced section will depend on magnitude of E_{SP}, the duration of the optical excitation pulse, and θ. It should be noted, however, that direct photoemission from the metal surface may influence the gating process via space-charge interaction. To circumvent the photoemission and eliminate the possibility of space-charge interaction, the surface of the metal can be engineered in such a way so as

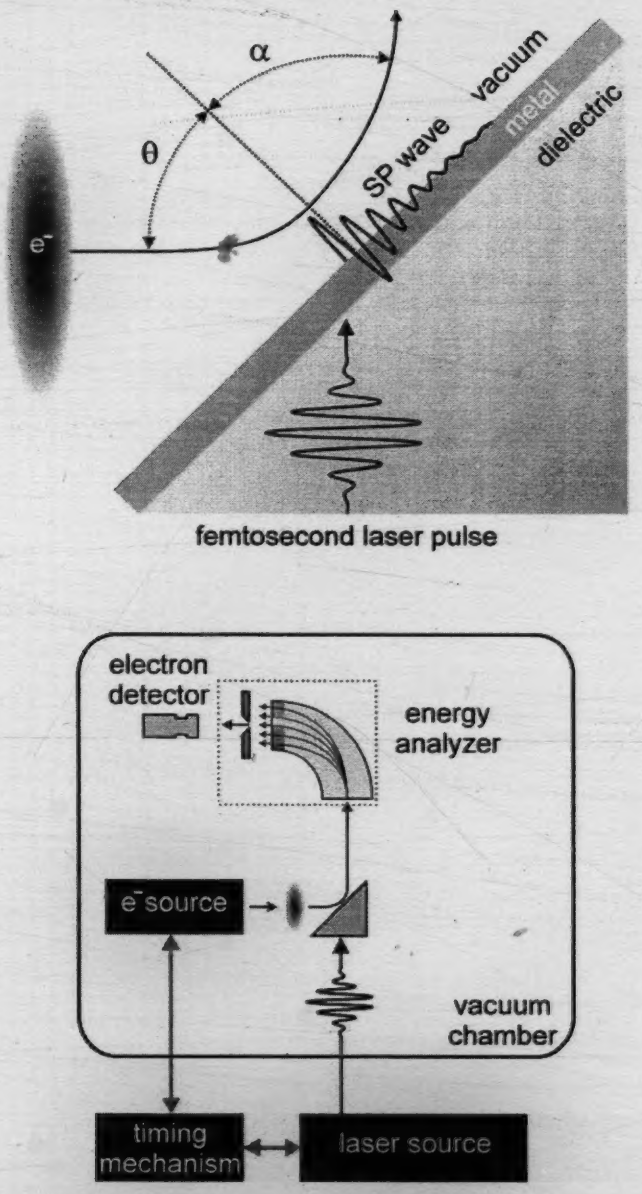

Figure 5.1. Arrangement for electron pulse gating using SP waves. (top) An external electron beam is directed toward a metal-coated prism surface at angle θ, measured from the surface normal. The electrons comprising the input beam are deflected and depart the interaction region at an angle α. Varying the delay between the launching of the SP and an incident electron packet allows the same SP-gating mechanism to be utilized for temporal characterization of electron pulses. (bottom) Potential experimental arrangement for realizing electron beam gating using SP waves, which consists of a laser source, an electron source, a timing mechanism to synchronize the optical and electron pulses, and an electron spectrometer for energy discrimination.

to inhibit photoemission processes, while maintaining E_{SP}. For example, an ultrathin large band-gap material can be deposited over the metal to prevent multiphoton electron emission.

For a finite duration electron packet, the relative delay between the electron beam and optical pulse becomes an important parameter in determining the efficiency and selectivity of the optical deflection mechanism. The relationship between the deflection efficiency and relative delay between the optical and electron pulses can be applied to ultrafast electron pulse duration measurement. In analogy to optical-optical correlation for the measurement of ultrafast laser pulses, electron-optical cross correlation allows for temporal characterization of electron pulses. For electron-optical cross correlation, the deflected charge is a function of the relative delay, τ, between the optical and electron pulses. Depending on the magnitude of the E_{SP}, electron deflection will be conditional in nature: the electrons are either deflected or not deflected from the metal surface. Hence, the SP wave can be described by a window function of the form

$$\Pi(t) = \Theta(t + t_0/2)\Theta(t_0/2 - t), \tag{5.1}$$

where $\Theta(t)$ is the Heaviside step function and t_0 is the width of the rectangular window. For illustrative purposes, a Gaussian form is assumed for the temporal envelope of the SP wave of width τ_p. Thus, the functional form of the width of the window function is given by

$$t_0 = \frac{\tau_p}{2\sqrt{\ln(2)}}\sqrt{\ln\left(\frac{I_0}{I_{th}}\right)}. \tag{5.2}$$

I_0 is the peak intensity of the SP wave and I_{th} is the minimum intensity required for ponderomotive electron deflection and is given by $I_{th} = 4m_e\omega^2 K_0/e^2$, where K_0, m_e, and e are the initial kinetic energy, mass, and charge of the electron, respectively. For a relative delay, τ, the deflected charge can be described through the cross-correlation function

$$Q_{deflected}(\tau) \propto \int_{-\infty}^{\infty} \Pi(t)\rho_{elec}(t+\tau)dt, \tag{5.3}$$

where ρ_{elec} is time-varying charge density of the electron pulse.

Calculation of the deflected charge $Q_{deflected}(\tau)$ requires knowledge of the local spatial and temporal distribution of the electric field near the metal-vacuum interface, which is accomplished by implementing the model described in Chapter 3. However, in the present situation, the electrons are directed toward the metal film from an external source. To determine the behavior of the electron pulse in the presence of the SP wave, a probabilistic calculation is implemented in which all possible electron paths are considered. Thus, the incident electron beam/pulse is represented by ~10^5 test electron trajectories, which are assigned relative weights to account for the packet's finite spatial extent and temporal duration. The trajectory of each test electron is, once again, determined through the classical Lorentz force equation.

The investigation of the SP-gating process was carried out using the same parameters described in Chapter 3. The excitation laser pulse has a central wavelength of λ_0=800 nm, τ_p=30 fs, and an incidence angle of 45°, while the metal parameters are taken to be those of a silver film: d=50 nm, ω_p=5.73×10^{15} Hz, and ν_d=1.3×10^{14} Hz [2].

The spatial step sizes of the computational lattice are chosen to be $\Delta x = \Delta y = d/10 = 5$ nm with a corresponding temporal step size of $\Delta t = 5\times10^{-3}$ fs. The zero time is defined to be the peak of the excitation pulse striking the metal film and all absolute time values are given with respect to this temporal origin.

5.1.2 Model Results for Electron Beam Slicing

The injection of electrons into the E_{SP} field and the subsequent ponderomotive photo-acceleration is a complex process. Several aspects of the deflected electron packets (e.g. angular and kinetic spectra, duration, etc.) are dependent on many variables including the magnitude of E_{SP}, the pulse duration of the excitation optical pulse, the incident angle, θ, and the specific time and location of electron entry into the SP wave. The following analysis elucidates the interaction of the incoming electron beam with the E_{SP}. Several aspects of the deflected electrons are investigated, which include: the kinetic energy spectra, the angular distributions, angle resolved spectra, and spatial and temporal distribution.

To illustrate the ponderomotive deflection process, several test electrons are directed towards the metal surface during the excitation of E_{SP}. Figure 5.2 illustrates representative trajectories of five $K_0 = 1$ keV electrons launched at various times of τ_e =-12, -6, 0, 6 and 12 fs with respect to the peak of the laser pulse. The test electrons are incident at $\theta = 45°$ and their paths are mapped as they traverse an E_{SP} that has a magnitude of 7.4×10^9 V/cm. As the electrons approach the SP field, their initial constant velocities are significantly modified as evidenced by their 'quivering' motion. It is

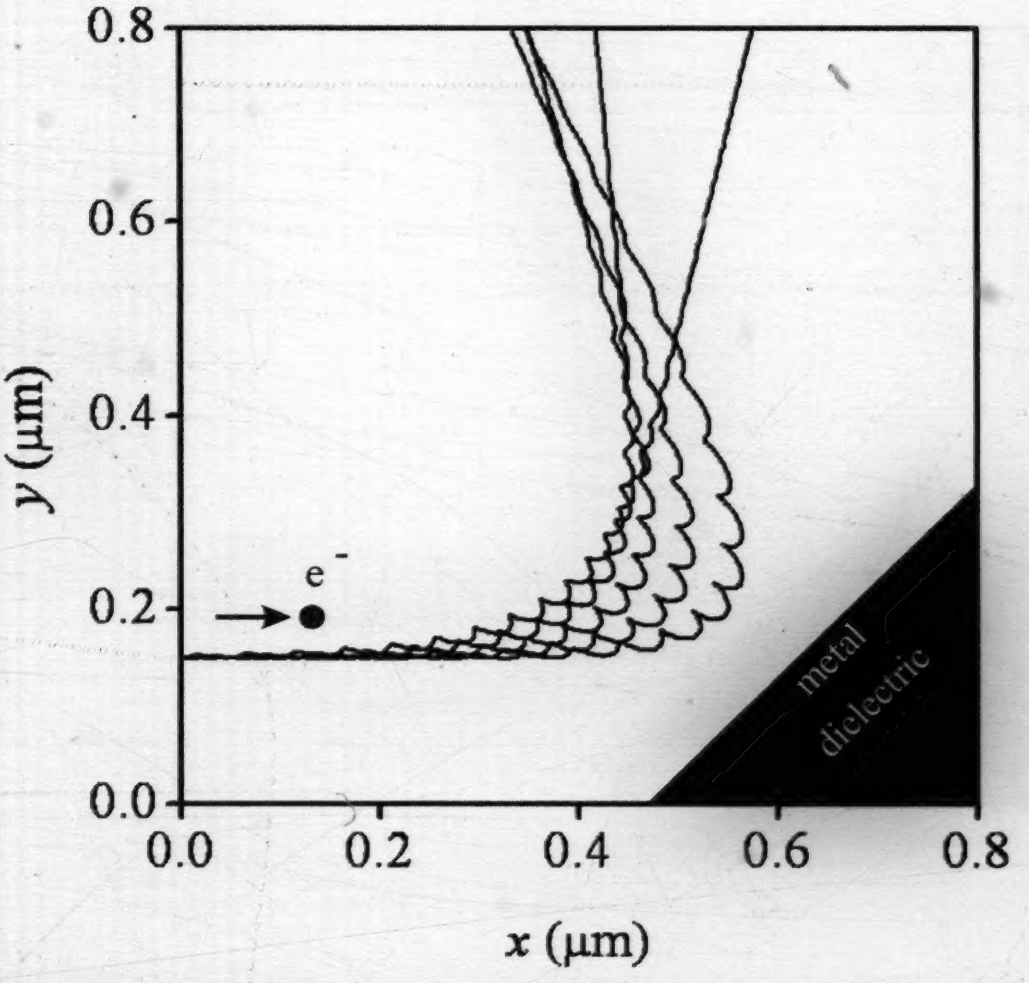

Figure 5.2. Trajectories of five test electrons as they interact with an SP wave having a peak electric field amplitude of E_{SP}=7.4×10^9 V/cm. The test electrons are delayed with respect to the peak of E_{SP} with launching times of τ_e = -12, -6, 0, 6 and 12 fs.

apparent that the electrons experience a time-average ponderomotive force in the direction of largest field gradient $\nabla|E_{SP}|^2$. The pertinent parameter in determining whether an electron is deflected away from the surface is its velocity component along the film's normal, $v_\perp$, or more specifically, the kinetic energy $K_\perp = m_e v_\perp^2/2$ associated with this velocity component. Once an electron decelerates and reaches a critical point within the E_{SP} field, where $K_\perp$ balances the ponderomotive potential of the SP wave, U_{SP}, the instantaneous $v_\perp(t)$ component will be reduced to zero and the electron has only a velocity component parallel to the film's surface, $v_\parallel$. Eventually, $v_\perp(t)$ will increase along the film's normal as the electron is pushed away from the film surface.

When the SP wave is excited with an ultrashort optical pulse, electrons on the leading or trailing edge of the wave will experience a dynamical ponderomotive potential that depends on the specific arrival time of the electron. As a result, the angle, α, through which the electron is deflected will vary with τ_e. Figure 5.2 illustrates that electrons can exit the surface with both $\alpha > \theta$ and $\alpha < \theta$, which correspond to the cases of kinetic energy loss or gain, respectively. Given that $\nabla|E_{SP}|^2$ is along the film's normal, only the electron velocity component along this direction, $v_\perp(t)$, can be substantially altered through the interaction. The condition $\alpha < \theta$ necessitates that the electron exits the interaction region with a kinetic energy $K > K_0$, where K_0 is its initial energy. This is due to the fact that the electron enters a location over which the SP extends spatially, but at a time *before* the peak excitation of the plasmon. Even though the electron is within the evanescent penetration depth of the SP wave, this electron is allowed to 'sample' a ponderomotive potential that is greater than the minimum potential required for

deflection. The opposite situation can also occur in which electrons enter the evanescent field during the trailing edge of the SP wave and suffer a reduction in $v_{\perp}(t)$. This is evidenced by the trajectory having the largest delay of 12 fs, illustrated in Figure 5.2, which exits the interaction with $\alpha > \theta$.

The effective ponderomotive potential, and hence the amount of energy transferred to the impinging electron, is a function of the duration of the optical excitation pulse. To determine the effective U_{SP} of the SP wave as a function of angle of incidence, electrons are directed toward the metal film surface at various θ. By incrementing E_{SP} for a given θ, the threshold electric field required for electron deflection can be measured and in turn, the effective ponderomotive potential of the SP wave can be determined. The situation is clearly depicted in Figure 5.3a-f for various θ ranging from 0 to 75°. Individual $K_0 = 1$ keV test electrons are direct towards the metal film at peak excitation of the plasmon, which ensures that the maximum surface field will be sampled. The threshold electric field, E_{SP}^{TH}, required for electron deflection, is defined as the minimum E_{SP} required such that the electron trajectory does not cross the metal-vacuum boundary. Each of the panels in Figure 5.3 illustrates electron trajectories for field values above and below such threshold values. It should be noted that, for $U_{SP} < K_{\perp}$, the electron's traced path crosses the plane of the film surface and is absorbed, however, the trajectory is shown for illustrative purposes. Conversely, if E_{SP} is above the threshold required for deflection, $U_{SP} > K_{\perp}$ will ensure that the electron has its $v_{\perp}(t)$ component altered such that the electron is deflected away. For each case that the electron is deflected, its new velocity component along the film normal is greater than or equal to its

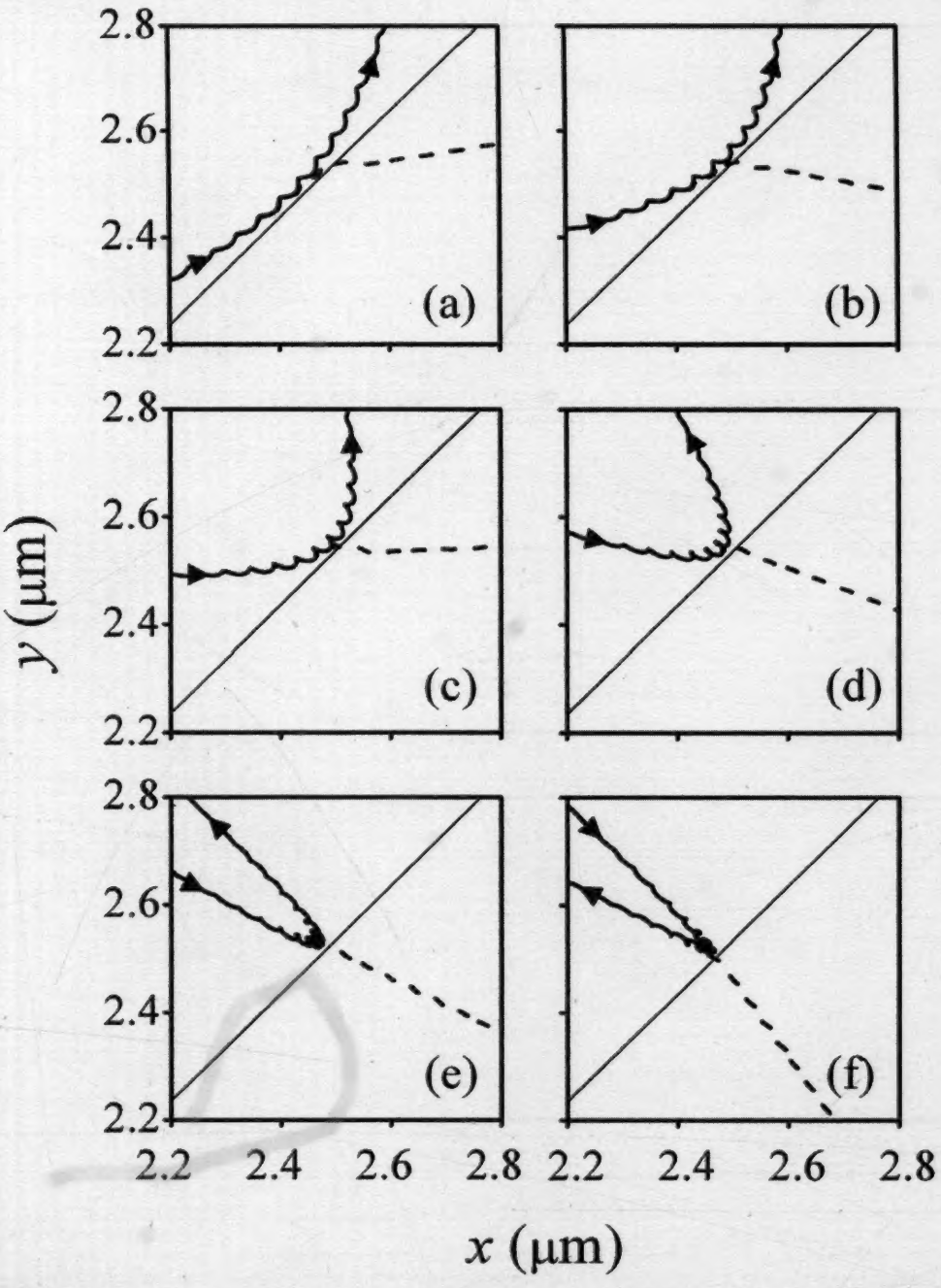

Figure 5.3. Electrons interacting with the SP wave for various θ of (a) 75°, (b) 60°, (c) 45°, (d) 30°, (e) 15°, and (f) 0°. For each panel, two electron trajectories are plotted corresponding to the cases of $K_{\perp} < U_{SP}$ (solid blue) and $K_{\perp} > U_{SP}$ (dashed red). The arrows indicate the direction of the electrons as they approach and exit E_{SP}.

initial velocity along the film normal. Notably, at such E_{SP}^{TH}, the deflection angles $\alpha \sim \theta$-12°. For $\theta = 0°$, however, the difference between the deflected angle and the incident angle is the largest at $\theta = 17°$. This effect is due to the finite wavevector of the SP wave: as the SP wave propagates along the metal film, the direction of largest gradient, as observed by the electron, acquires a slight tilt with respect to the film surface and preferentially forces the electron to move along this direction. The results shown here indicate the potential of using electrons and their subsequent deflection to probe the magnitude of the surface electric field, which currently, is a challenging parameter to measure.

In a quasi-static situation, where the E_{SP} field is turned on for a long period of time (i.e. longer than the interaction time between the electron and E_{SP}), the ponderomotive potential can be calculate using $U_{SP} = q^2 E_{SP}^2 / 4m_e\omega^2$ [3]. However, the plasmon field is generated from an ultrashort pulse and therefore the effective ponderomotive potential will be reduced (see discussion in section 3.3). Moreover, electrons can be incident at arbitrary angles and the effective ponderomotive potential will vary accordingly with θ. The two effects can be incorporated into a modified equation describing the threshold value of ponderomotive potential

$$U_{TH} = \beta \frac{q^2 \left(E_{SP}^{TH}\right)^2}{4m_e\omega^2} \cos^2\theta, \tag{5.4}$$

where β is a constant accounting for the finite duration of the SP wave [3]. Figure 5.4 illustrates the simulated U_{TH} as a function of θ as calculated from the E_{SP}^{TH} values. Good agreement between the model calculations and equation 5.4 is achieved with $\beta = 2.1$,

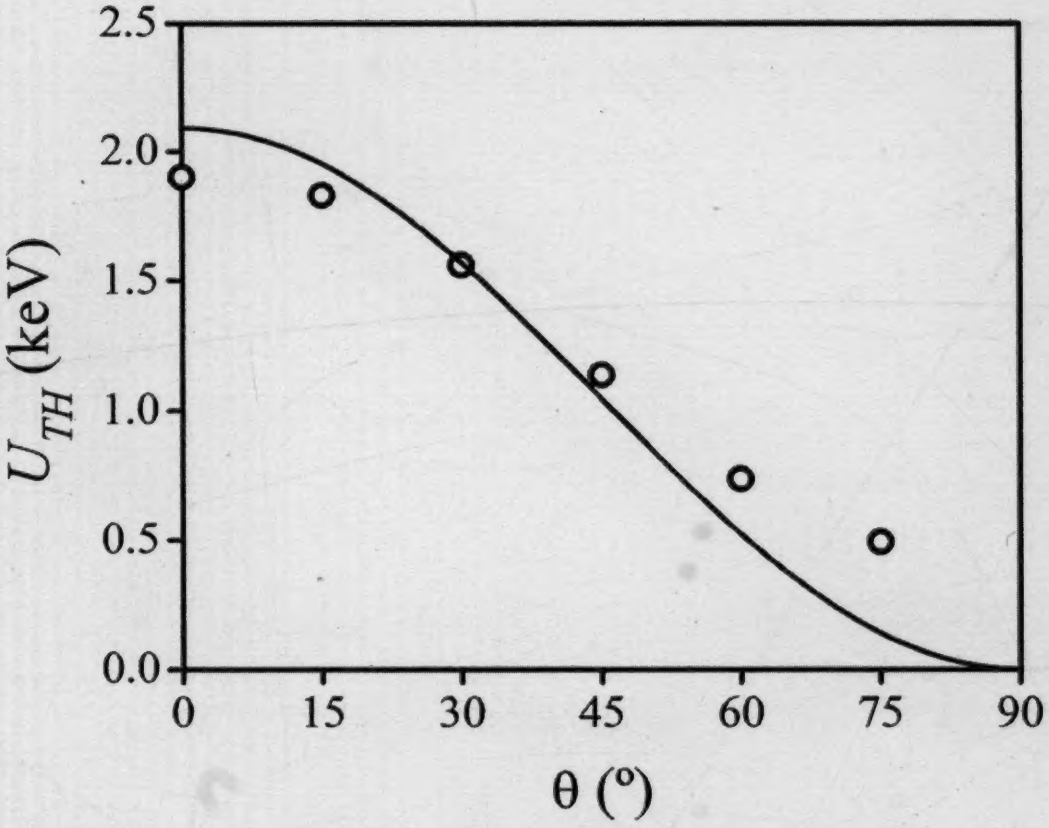

Figure 5.4. Comparison of threshold values of the ponderomotive potential required for electron deflection as calculated from the model (circles) and equation 8 (solid line).

indicating that the effective ponderomotive potential is reduced by over a factor of 2 as compared to the quasi-static value. A distinct trade-off emerges in that a longer optical pulse can be used to lower the required U_{TH}, however, at the expenditure of increasing the duration of the deflected electron packet. It is also important to note the significantly reduced U_{TH} near 90°. To avoid the generation of electrons via photoemission, θ can be sufficiently large (~ 90°) such that the laser beam intensity is reduced to ≤ 1 GW/cm^2, which in turn, would reduced the number of photoelectrons produced at the metal surface.

To characterize a femtosecond electron packet generated via the SP-gating process, a continuous beam of electrons is directed toward the metal film with $\theta = 45°$ and K_0 = 1 keV. Several snap-shots at times ranging from t = -20 fs to 130 fs are shown in Figure 5.5, illustrating the result of the interaction between the incoming electron beam and the SP electric field of E_{SP}=7.4×10^9 V/cm. Initially (t < -20 fs), electrons incident upon the metal are not deflected, as E_{SP} has not reached its peak value. These electrons can either reflect off the surface of the metal film or generate secondary electrons. In either case, such electrons are not synchronized with the laser pulse and would appear as a DC offset in the deflected current. Once the optical pulse couples to the SP wave ($t = 0$), the ponderomotive force exerted by E_{SP} deflects electrons and a significant portion of the original electron beam has been sliced and redirected away from the prism surface.

Further examination of the spatial distribution of the deflected electron pulse indicates a high degree of spatial microbunching, with an average distance between peaks

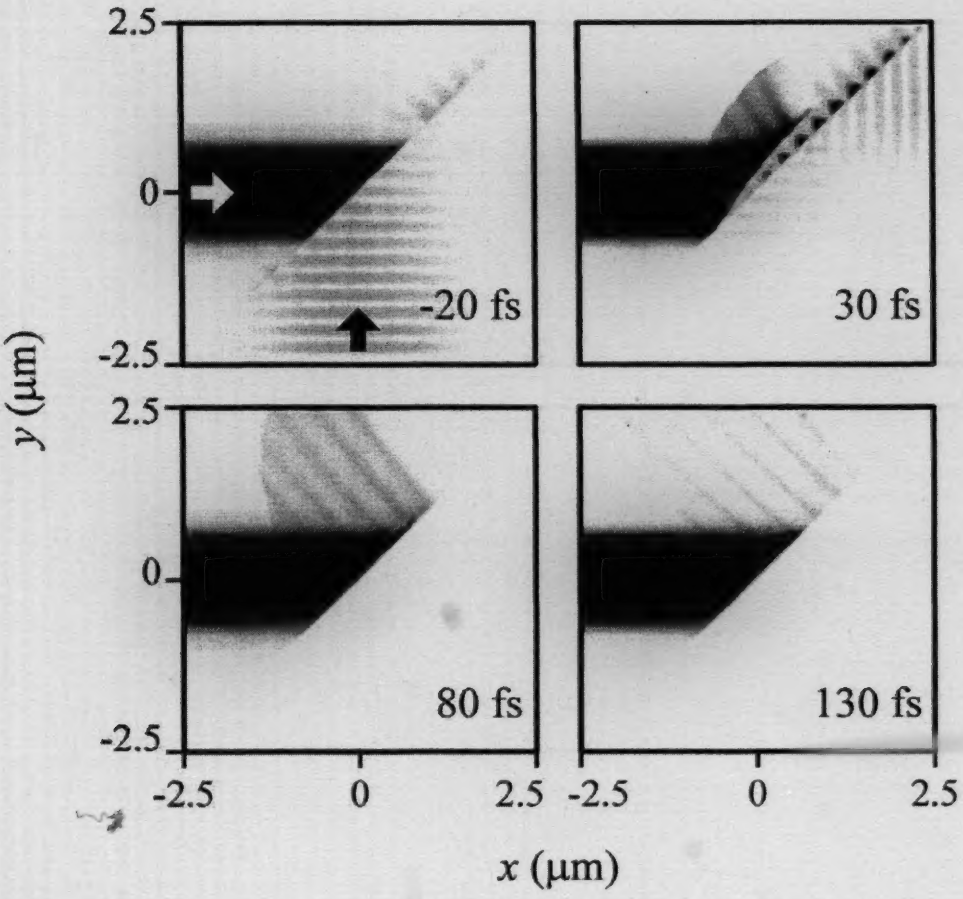

Figure 5.5. Snapshots of the SP-gating of an electron beam at various times ranging from -20 fs to 130 fs. The white arrow indicates the direction of the wavevector of the incident laser pulse, while the black arrow indicates the propagation direction of the electron beam.

equal to half the wavelength (~400 nm) of the optical excitation pulse. To observe the temporal characteristics of the microbunches, the electron packet is sampled and illustrated in Figure 5.6 for five sampling locations both perpendicular and parallel to the metal surface. For detection along the perpendicular direction, each curve of Figure 5.6a has the same overall pulse shape comprising a fast rise of ~10 fs and a slow fall of ~72 fs with no indication of microbunching. As the detector distance from the metal surface is increased, the full-width at half-maximum (FWHM) of the electron density curves increase from 34 to 43 and to 44 fs for distances of 1.0, 1.5, and 2.0 μm, respectively. Figure 5.6b illustrates the variation of the FWHM with distance along a direction 22° away from the surface normal (see the inset of Figure 5.7a). These results indicate that the deflected electron packet's energy distribution is non-monoenergetic. Furthermore, as the distance of the detector increases, the amplitude of each curve decreases, suggesting that the electrons comprising the packet do not depart the surface at the same α. The variation of total number of deflected electrons with distance is shown in Figure 5.6c along the direction 22° away from the surface normal. However, as shown in Figure 5.6d, the measured temporal profiles parallel to the film surface reveal broad envelopes (160 fs) with an underlying waveform composed of eight ultrashort packets corresponding approximately to the number of electric field oscillations of E_{SP}. The average duration of these subsidiary pulses is 13 fs, and span the range from 7 to 23 fs. Interestingly, the packets' durations increase with time, signifying that the duration of the underlying packets are a function of the time spent by the electrons in E_{SP}. It is interesting to note that, even though there is no velocity matching between the electrons and the SP wave,

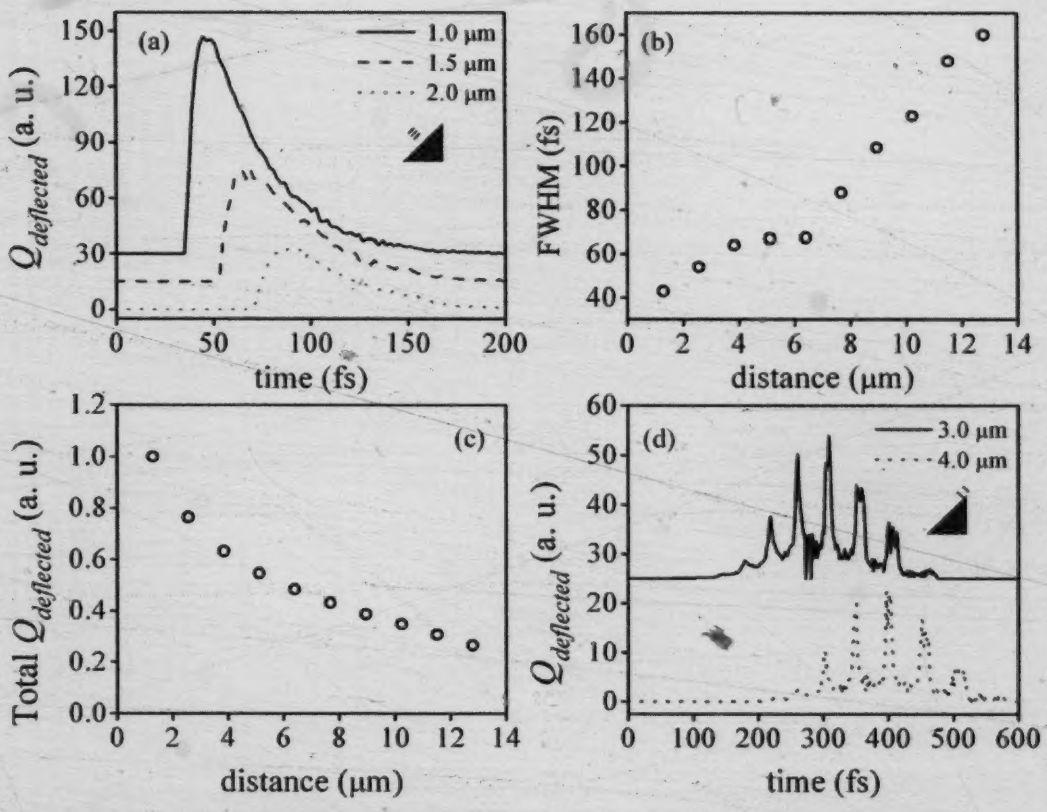

Figure 5.6. (a) Number of deflected electrons as a function of time at various sample locations at perpendicular distances of 1.0, 1.5, and 2.0 μm away from the metal film surface. (b) Variation of the FWHM of the electron packet as a function of distance away from the prism surface along a direction 22° from the surface normal. (c) Variation of the total number of deflected electrons as a function of distance away from the prism surface along a direction 22° away from the surface normal. (d) Number of deflected electrons as a function of time at two locations along the film surface for distances of 3.0 and 4.0 μm away from the center of the metal surface. Legends in (a) and (d) contain illustrations depicting the location of the detector with respect to the prism surface. It should be noted that in (a) and (d) the curves have been offset vertically for clarity.

electrons acquire a preferential spatial distribution from electromagnetic fields of the plasmon.

The energy spectrum of the sliced electron pulse is shown in Figure 5.7a. Here, it is observed that the initial mono-energetic spectrum of $K_0 = 1$ keV electrons has broadened significantly following its interaction with the SP field. The large peak occurring at 330 eV with a FWHM of 179 eV indicates that significant portions of the electrons lose energy during the deflection process. However, an approximately equal number of electrons have gained energy, up to 4 keV, from E_{SP}. Of particular interest is the angular distribution shown in the inset of Figure 5.7a, which illustrates the number of deflected electrons as a function of exit angle, α. The highly directional nature of the deflected packet is observed with a peak at 22° and an angular half-width of 21°. Interestingly, a significant number of electrons have final velocity vectors parallel to the film surface, indicating that some electrons are decelerated to the critical point where $v_\perp \sim 0$. The directional characteristic of the sliced electron packet can be utilized to spatially filter the deflected electron beam and discriminate against background electrons generated from photoemission. Moreover, angle-resolved spectra shown in Figure 5.7b clearly exemplify the large correlation between final energy and direction of the deflected electrons. A closer examination of Figure 5.7b reveals seven distinct energy bands indicating that the electrons are grouped with respect to their energies. As α approaches 90°, the energy bands asymptotically converge to values near 330 eV as shown in Figure 5.7a. The overall characteristic of these energy bands can be determined by considering that the ponderomotive force is nearly parallel to the normal of the metal film surface. Through simple analysis and assuming that $dv_\parallel / dt \approx 0$, the dependence of final energy on

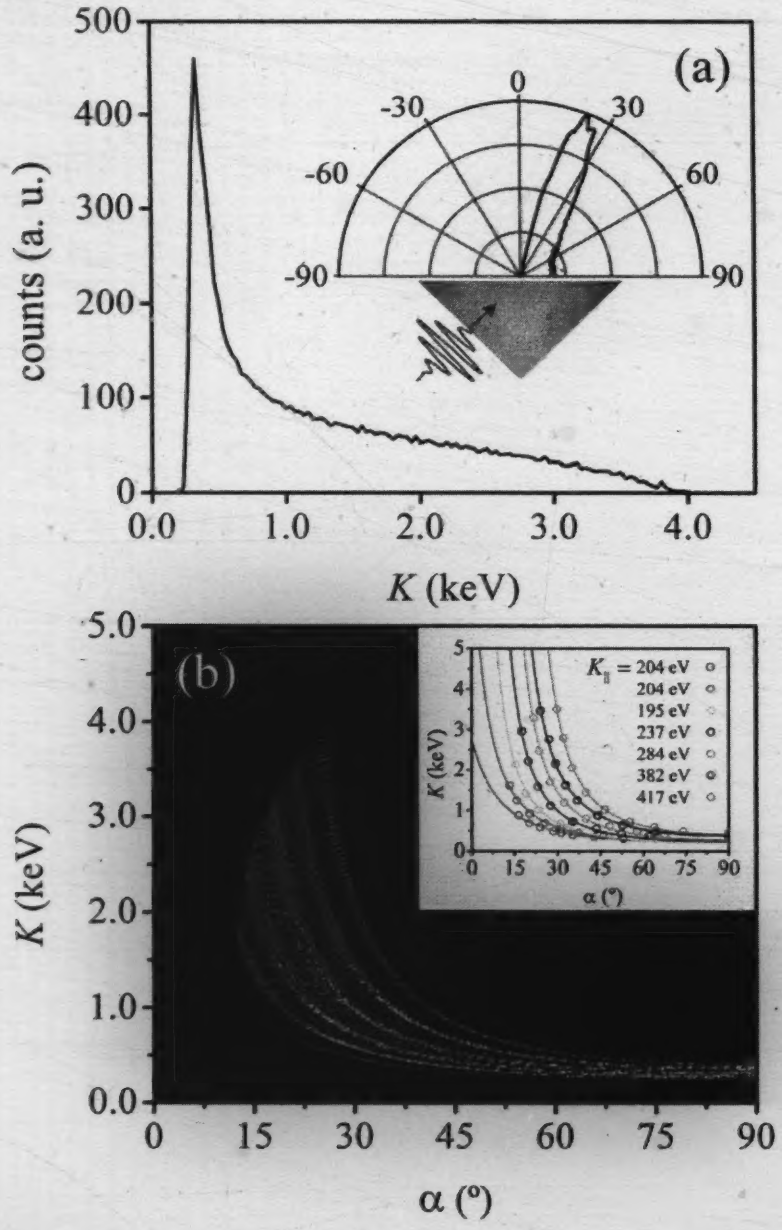

Figure 5.7. (a) Kinetic energy spectra of the sliced electron beam depicted in Figure 5.6. The peak has a central value of 330 eV and a FWHM of 179 eV. The inset shows the angular distribution of the sliced electron beam and its relation to the surface of the prism. Directionality of the sliced beam is evidenced by the peak at 22°, which has an angular half-width of 21°. (b) Angle-resolved energy spectra reveal distinct energy bands that follow $K \sim K_{\parallel}\left(1+\cot^2\alpha\right)$ for various $K_{\parallel}$ values ranging from 204 to 417 eV.

outgoing angle can be derived to be $K \sim K_{\parallel}\left(1+\cot^2\alpha\right)$ for an individual energy band. Good agreement between K, calculated from this equation, and the model energy bands is shown in the inset in Figure 5.7b. The dependence of final energy on outgoing angle provides a method for separating the electron bunches, which is extremely important for experiments that require electron pulses with an ultrashort duration and narrow energy bandwidth. Experimentally, the electron bunches can be separated according to their energy by employing an electron energy analyzer (shown in Figure 5.1) having a finite acceptance angle.

To demonstrate that the aforementioned technique can be employed for electron-optical cross-correlation, the electron beam is replaced with an ultrashort electron pulse. Here, the deflected charge is a function of the relative delay, τ, between the optical and electron pulses. The situation is clearly depicted in Figure 5.8 which illustrates a $K_0 = 1$ keV, 50 fs electron pulse interacting with the SP electric field of $E_{SP} = 7.4\times10^9$ V/cm for $\tau = 0$. The deflected electron pulse demonstrates much of the same behavior as the electron packet sliced from the previously discussed electron beam, with one important difference: the deflection efficiency is now a function of the temporal overlap of the electron pulse with the SP wave. Figure 5.9 illustrates the cross-correlated deflected charge, $Q_{deflected}(\tau)$, as a function of τ calculated for various E_{SP}. The overall shapes of each curve are Gaussian with FWHM of 72, 90, and 100 fs for E_{SP} of 3.7×10^9 V/cm, 7.4×10^9 V/cm, and 1.9×10^{10} V/cm, respectively. As described previously, the width of the cross-correlation function depends on the intensity of the SP wave and is verified by the increase of the FWHM with increasing E_{SP}. Furthermore, the amount of $Q_{deflected}(\tau)$

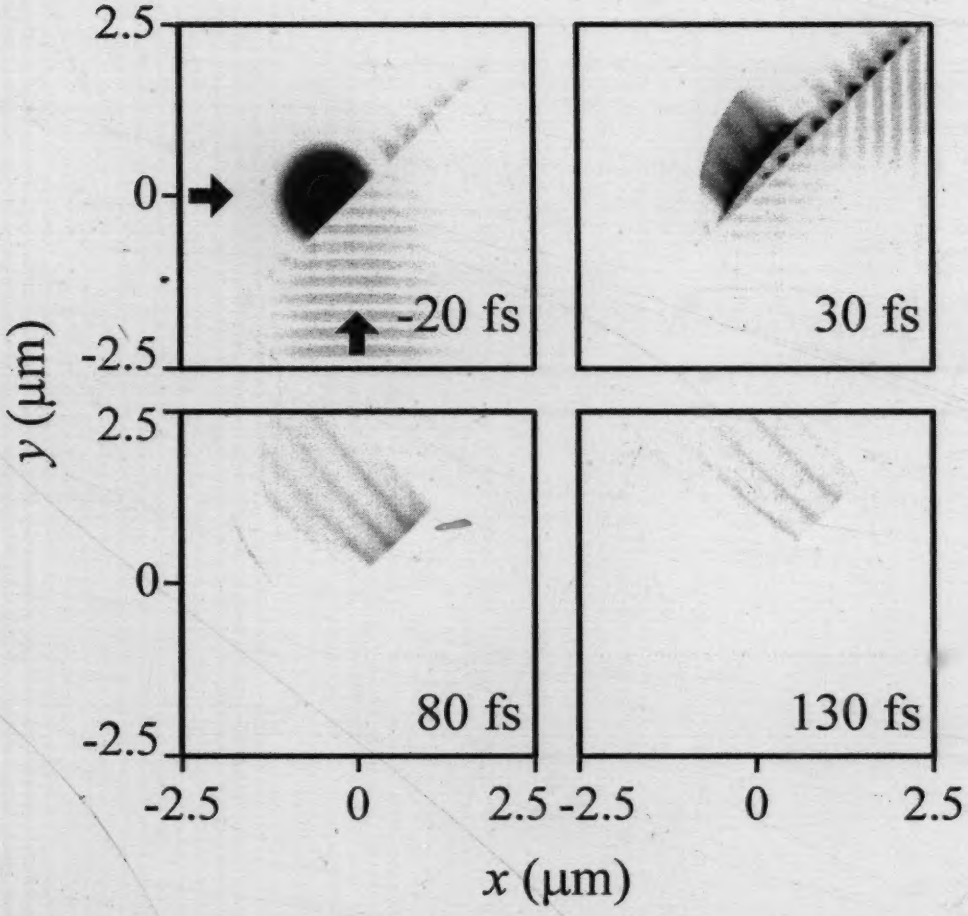

Figure 5.8. Snapshots of the SP-gating of an electron pulse at various times ranging from -20 fs to 130 fs for a relative delay of $\tau = 0$. The arrows indicate the direction of the propagation of the electron and optical pulses.

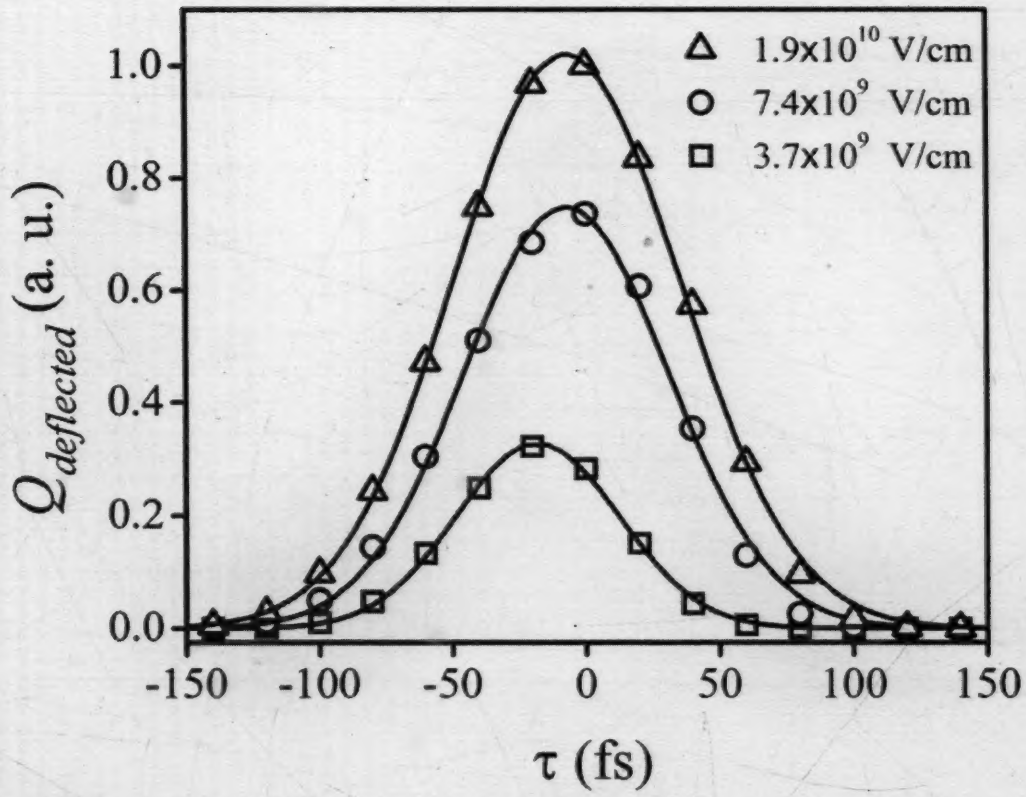

Figure 5.9. Cross-correlation between an SP excited with a 30 fs optical pulse and a 50 fs electron pulse for various E_{SP} of 3.7×10^9 V/cm, 7.4×10^9 V/cm, and 1.9×10^{10} V/cm.

is also a function of E_{SP}, where an increase of E_{SP} results in an increase of the amplitude of the correlation function. For the largest $Q_{deflected}(\tau)$ shown in Figure 5.9, 33% of the incoming electrons have been deflected. Complete deflection of the incident electron pulse can be achieved by choosing E_{SP} sufficiently large ($>> E_{SP}^{TH}$), however, in such a case no inference can be made of the electron pulse duration. Conversely, increased temporal resolution can be achieved at the cost of a diminishing deflected signal level (see Figure 5.9). Thus, a trade-off between the temporal width of the correlation and the amount of deflected charge emerges. An electron pulse representing a delta-function can be used to establish the resolution of the cross-correlation process. Figure 5.10 illustrates several cross-correlations corresponding to a 30 fs optical excitation pulse and various electron pulses having durations ranging from 5 to 200 fs. As the duration of the electron pulses are reduced, the FWHM of the curves approach a constant value of 77 fs corresponding to the temporal convolution width. Since the ponderomotive interaction is purely electromagnetic (i.e. all-optical), the minimum obtainable resolution is limited essentially by how fast the SP wave can be 'turn on' and 'turn off'. For the present analysis, silver metal film parameters were chosen and as such the plasmon lifetime is $\tau_{plasmon} = 48$ fs [4]. By choosing a metal film having a lower lifetime (e.g. gold), the resolution of the optical-electron cross correlation can be increased even further, thus offering the unique and exciting possibility of electron pulse characterization on timescales below 10 fs.

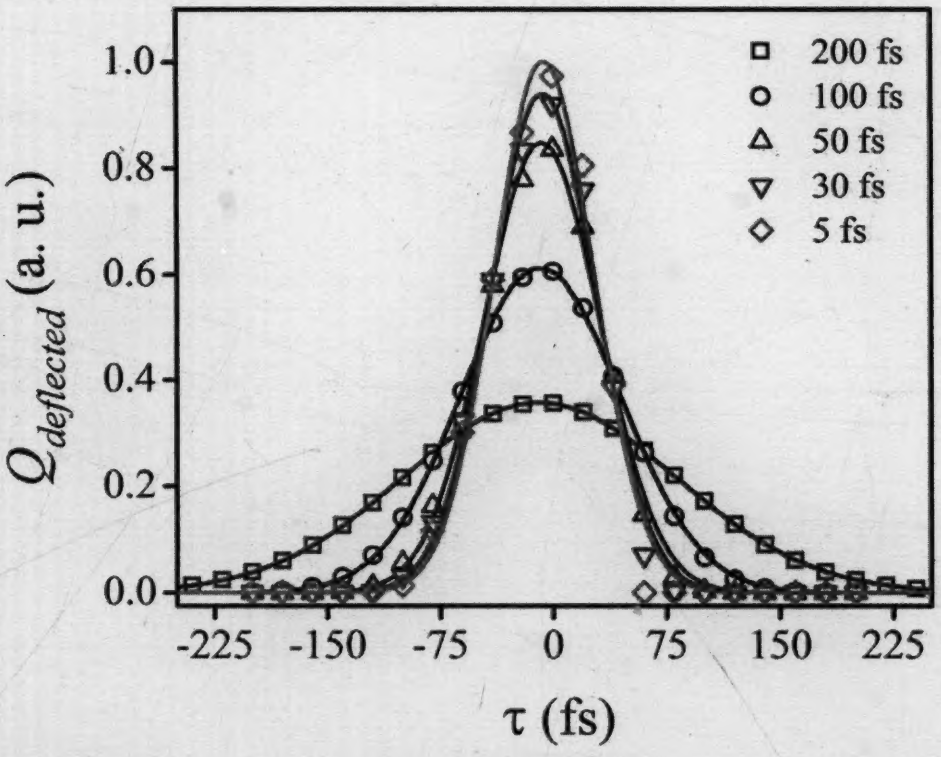

Figure 5.10. Cross-correlation between an SP excited with a 30 fs optical pulse and electron pulses with durations ranging from 5 to 200 fs. The resolution of the system determined from the 5 fs electron pulse is 77 fs.

5.2 Influence of Carrier-Envelope Phase on SP Electron Acceleration

An ultrashort few-cycle laser pulse can be characterized by an electric field of the form $E_L(t,\varphi_{CEP}) = E_0(t)\cos(\omega t + \varphi_{CEP})$, where $E_0(t)$ is the temporal envelope of the laser pulse, ω is the carrier frequency, and φ_{CEP} is the carrier-envelope phase (CEP) of the electric field oscillation relative to the envelope peak $E_0(t=0)$. In the most general situation, optically driven processes lack sensitivity to φ_{CEP} as $\tau_p >> T_0$, where τ_p is the duration of the laser pulse and $T_0 = 2\pi/\omega$ is the period. As is shown below, the SP electron acceleration mechanism is no exception to this general rule. However, cases where $\tau_p \sim T_0$ provide the opportunity to study the nature of SP electron interaction on a timescale comparable to a single light-wave oscillation.

The geometry for *few-cycle* SP electron acceleration is shown in Figure 5.11 and is identical to those discussed in previous chapters: the few-cycle laser pulse is used to excite an SP wave at a metal-vacuum boundary through the Kretschmann configuration. More important, however, is the coherent impression of the CEP of $E_L(t,\varphi_{CEP})$ onto the temporal structure of the plasmon wave and its subsequent effect on charged-particle acceleration. Photoelectrons, produced at the metallic surface during the same instant that the SP is launched, will be accelerated to considerable energies by the ponderomotive force resulting from the high gradient E_{SP}. The ponderomotive gain experienced by an electron is contingent upon the instantaneous value of E_{SP} during its photo-injection and subsequent interaction; therefore, it is expected that the energies of the photo-accelerated electrons will be sensitive to the laser-parameter φ_{CEP} when $\tau_p \sim T_0$.

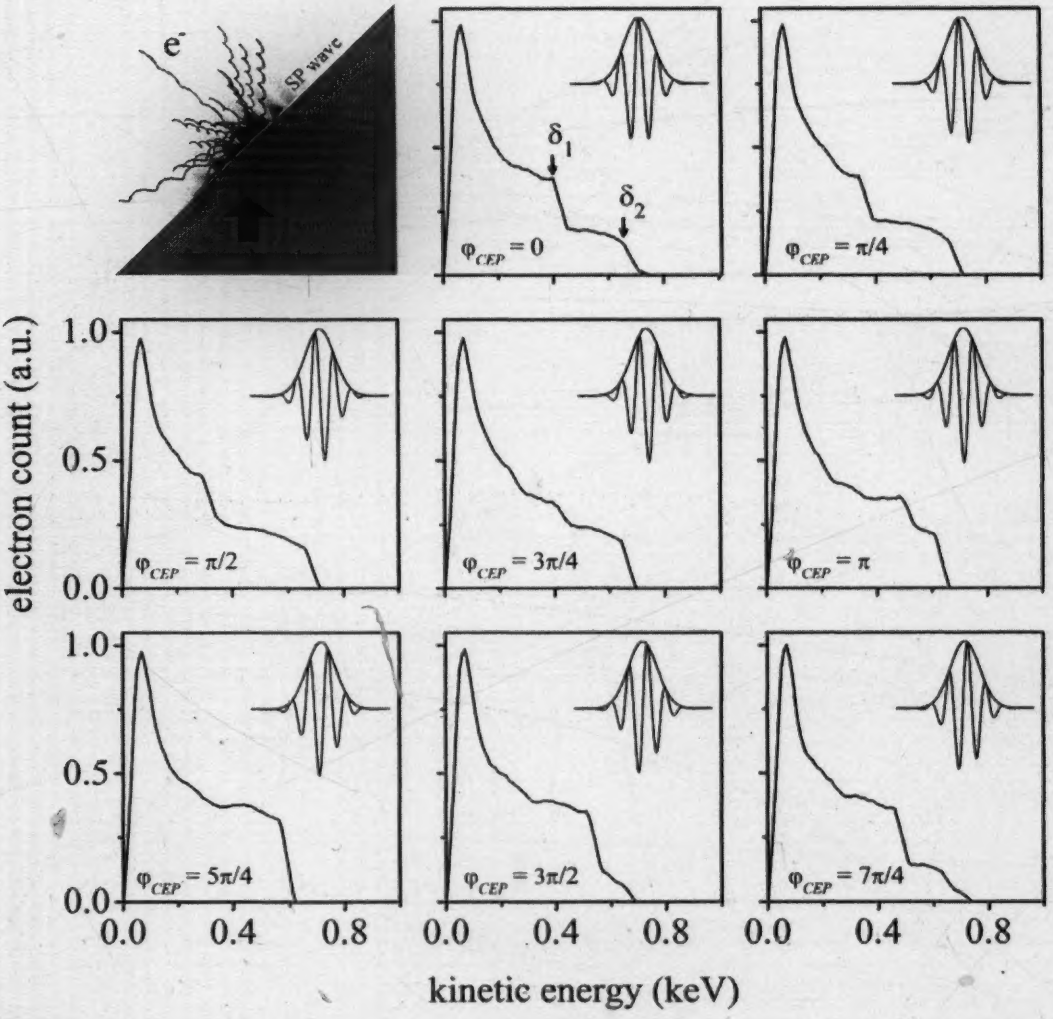

Figure 5.11. Upper left panel: illustration of the launching of an SP wave and subsequent dynamics of photo-injected electrons accelerated during the interaction with an SP wave excited with a τ_{laser}=5 fs. The succeeding panels depict the electron energy spectra at various φ_{CEP} ranging from 0 to 2π. For each value of φ_{CEP}, the insets show the electric field waveform, $E_L(t,\varphi_{CEP}) = E_0(t)\cos(\omega t + \varphi_{CEP})$, and its specific relation to the Gaussian optical pulse envelope. Two pronounced cutoffs, positioned at values of $\delta_1 = 425$ eV and $\delta_2 = 685$ eV, are evident within the energy spectra.

The model described in Chapter 3 is applied to study this phenomenon, however, additional consideration must also be given to the photo-ejection mechanism of the electrons of the metal film. It has been shown previously through density functional theory (DFT) that the photoemission process itself can, in fact, depend upon the value of φ_{CEP} for the regimes of $\gamma > 1$ or $\gamma < 1$ [5]. Marriage of such a DFT model with the electromagnetic description described in Chapter 3 of this thesis is possible, however, would require enormous computational effort as the two separate physical descriptions occur on completely different spatial scales (1 μm vs. 0.1 nm). On the other hand, recent experiments [6] attempting to verify the DFT model for $\gamma > 1$ showed only a small variation (<0.1%) of electron count with φ_{CEP}. Therefore, the first-order assumption that the electronic charge emitted by the laser pulse is independent of the underlying waveform and follows the intensity of the laser pulse is adopted. While this assumption would no longer be valid in the cases where $\gamma < 1$, experiments at laser-oscillator energies ((described in Chapter 4) indicate that multiphoton absorption is the dominant photoemission mechanism [7,8], and therefore, the discussion is restricted to the nonadiabatic (multiphoton) $\gamma > 1$ regime. Hence, the rate of photoelectron generation is proportional to $I_{laser}^{m}(x,y,t)$, where m is the order of the photoemission process. Again, the wavelength of the optical excitation pulse is $\lambda_0 = 800$ nm and the metal film parameters are taken to be those of silver (m=3).

Calculated energy spectra of SP accelerated electrons are shown in Figure 5.11 for $\tau_p = 5$ fs, φ_{CEP}=0 to 2π, and a peak E_{SP}= 1.8×10^{9} V/cm. Overall, each of the electron energy spectra span the range from 0 to 750 eV and contain a low energy peak

located at 70 eV. Two pronounced cutoffs, positioned at values of $\delta_1 = 425$ eV and $\delta_2 = 685$ eV for $\varphi_{CEP}=0$, are clearly evident within the energy spectra. The origin of δ_1 and δ_2 is directly associated with the acceleration mechanism. For adiabatic ponderomotive forces, acceleration takes place over many cycles of the SP wave, and a photo-injected electron is allowed to 'feel' many oscillations of the E_{SP}. Over time, the electron acquires a velocity that is proportional to the difference in the peak values of the subsequent electric field oscillations that the electron 'sees' as it interacts with the SP field [9]. That is, the electron energy gain is proportional to the gradient of the time-average $\left\langle E_{SP}^2 \right\rangle_t$. In such cases, where $\tau_p >> T_0$, the difference in neighboring peak electric field values is infinitesimal and translates into an equally incremental change in electron energy, ΔK. Depending on the time and location of emission into this field, an electron can accumulate a number of these discrete energy differences ranging from 0 to $n\Delta K$, where n is the number of electric field oscillations comprising the optical pulse. Since ΔK approaches zero for $\tau_p >> T_0$, the associated CEP effects will be insignificant. On the other hand, few-cycle SP acceleration is non-adiabatic in nature as ΔK is no longer infinitesimal. Strictly speaking, SP electron acceleration in a few-cycle regime does not allow for time averaging over many oscillations of E_{SP}, and thus, is not truly ponderomotive in nature. As the excitation optical pulse is delta-function like, the electrons accelerated by the resultant SP wave will bear a signature of the underlying phase since ΔK is much larger as compared to case of many-cycle pulses. The spectra of Figure 5.11 clearly exemplifies this situation. In this case $\tau_p = 5$ fs, there are essentially

only two periods ($n=2$) at $\lambda_0 = 800$ nm, which manifest themselves as δ_1 and δ_2 within the electron energy spectra.

To illustrate the dependence of energy of the SP accelerated electrons on the CEP, spectra having various values of φ_{CEP} are overlaid with each other and plotted in Figure 5.12a. While the spectra do not indicate any observable dependence on φ_{CEP} below energies of 200 eV, it is observed that the electron count above energy $K = 300$ eV (which represents ~36% of the energy spectrum) has a marked dependence on φ_{CEP}. As

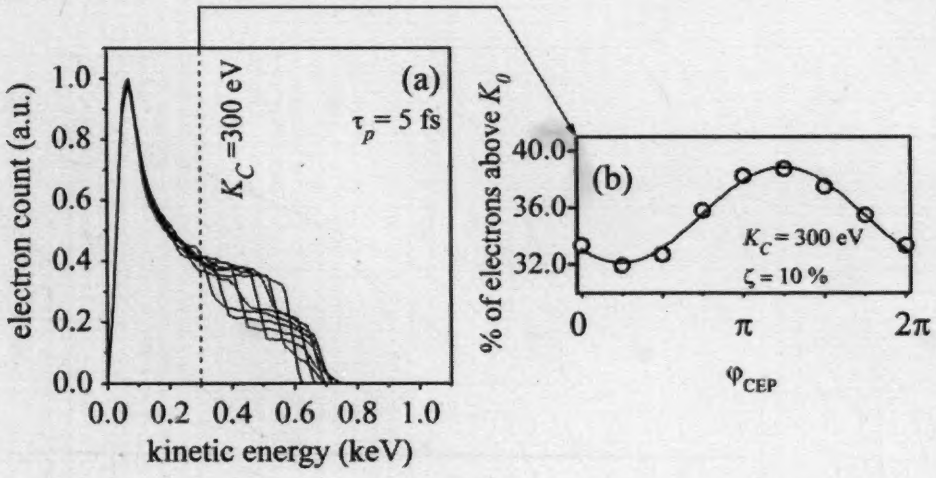

Figure 5.12. (a) Overlapped energy spectra of SP-accelerated electrons for φ_{CEP} ranging from 0 to 2π and $\tau_p = 5$ fs. (b) The variation of the total number of electrons above $K_C = 300$ eV, which is also indicated by a dashed line in (a).

illustrated in Figure 5.12b for an energy range $K > K_C$ (=300 eV), there is a clear sinusoidal relationship between the electron count, Q, and φ_{CEP}:

$$Q(K_0, \varphi_{CEP}) = A(K_C)\sin(\varphi_{CEP} + \varphi_0(K_C)) + Q_0, \tag{5.5}$$

where $\varphi_0(K_C)$ is the initial phase of the Q waveform for the energy range specified above K_C, $A(K_C)$ is the amplitude, and Q_0 is the baseline offset. Of particular interest is the contrast ratio, $\zeta = A(K_C)/Q_0$, which can be used as a figure of merit for the degree of CEP phase control. Shown in Figure 5.12b, a significant ζ=10% is measured, corresponding to a change of 7% of the total number of electrons within the spectra.

Up to this point, two-cycle laser pulses ($\tau_p \sim T_0$) have been considered; to further demonstrate the phase sensitivity of the SP acceleration process at longer pulse durations, τ_p is increased to 12 fs. Shown in Figure 5.13a are the calculated electron energy distributions generated using $\tau_p = 12$ fs for $\varphi_{CEP} = 0$ to 2π. Overall, each curve has the same characteristics of the spectra shown in Figure 5.12a. Since the longer τ_p allows more interaction time between the photo-injected electrons and E_{SP}, the peak and maximum energy have up-shifted by 20% [10]. Despite the fact that an optical pulse having $\tau_p \sim 4.5T_0$ is used the electron energy distributions still exhibit a significant φ_{CEP} dependence. Careful inspection of the energy distributions reveal $n \sim 5$ distinct regions where the electron count changes significantly with the CEP, matching approximately the number of the optical cycles in the 12 fs pulse. With the choice of a discrimination range $K_C = 720$ eV, the sinusoidal $Q(K_C, \varphi_{CEP})$ curve shown in Figure 5.13b is obtained. Again, a contrast ratio of up to 10% is realized. However, owing to the increased pulse duration (and hence, less pronounce CEP effects), this sinusoidal variation accounts for <1% of the total number of electrons comprising the spectrum. As expected, when τ_p is

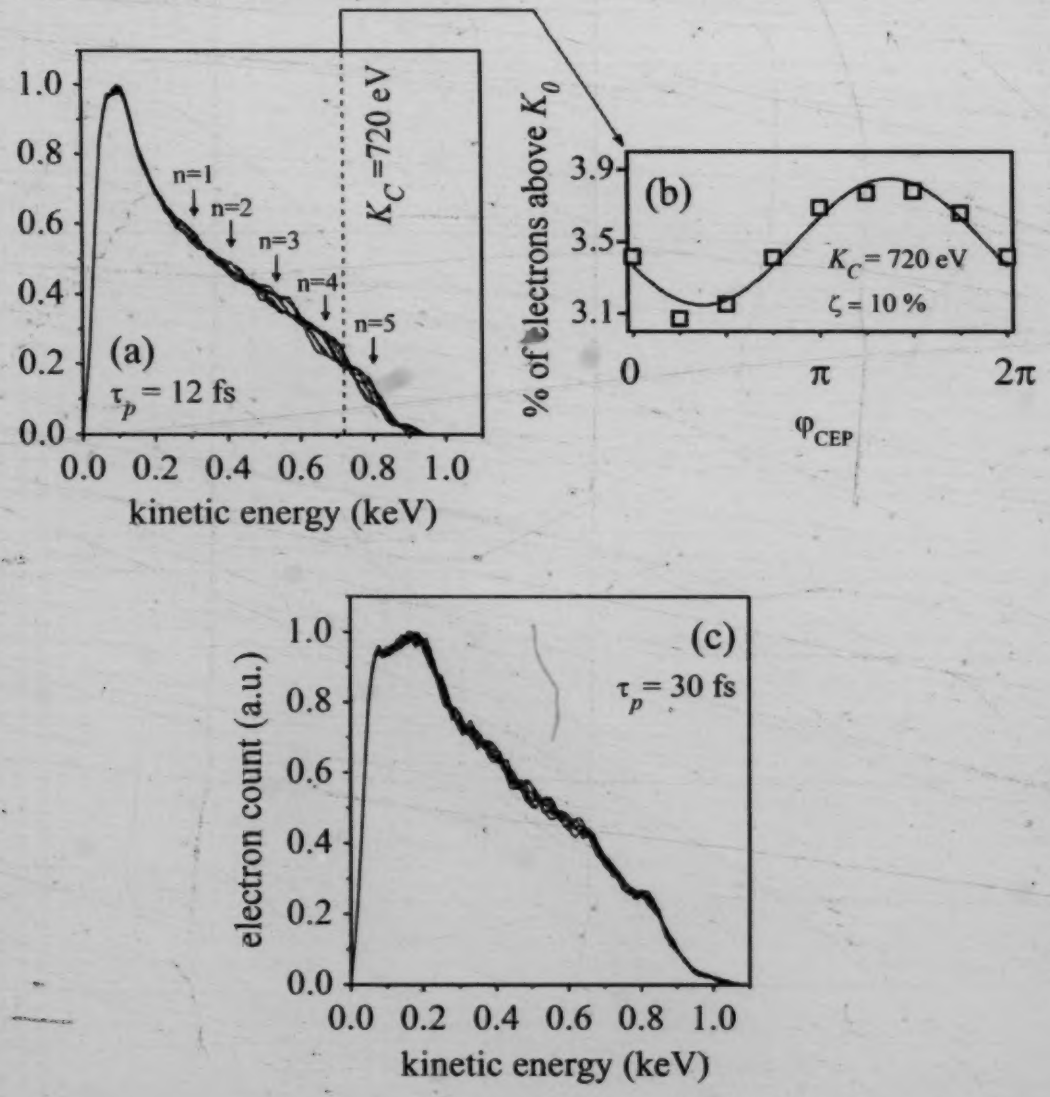

Figure 5.13. (a) Overlapped energy spectra of SP-accelerated electrons for φ_{CEP} ranging from 0 to 2π and $\tau_p = 12$ fs. The arrows in (a) indicate regions of CEP sensitivity. (b) The variation of the total number of electrons above $K_C = 720$ eV, which is also indicated by a dashed line in (a). Panel (c) illustrates overlapped energy spectra for $\tau_p = 30$ fs, which shows no indication of CEP effects.

further increased to 30 fs, all indications of φ_{CEP}-sensitivity vanish as evidenced by the indistinguishable overlapping electron energy spectra shown in Figure 5.13c.

To investigate the nature of the phase sensitivity and its relationship to electron energy, K_C is continuously varied across the entire energy spectra of the SP accelerated electrons. Figure 5.14a illustrates $\Delta Q(K_C,\varphi_{CEP}) = Q(K_C,\varphi_{CEP}) - Q_0$ and its variation with both K_C and φ_{CEP} for $\tau_p = 5$ fs. For a fixed φ_{CEP}, it is observed that $\Delta Q(K_C,\varphi_{CEP})$ remains relatively constant as K_C is varied from 0 to 300 eV. The lack of CEP sensitivity of the low energy electrons (<300 eV, see Figure 5.12a) is attributed to the fact that either these electrons do not spend sufficient time interacting with E_{SP} field and/or are injected near the wings of the SP wave [10,11]. In either case, the underlying CEP is not imprinted onto those particular low-energy electrons. However, for the energy range 300 eV $< K_C <$ 600 eV, $\varphi_0(K_C)$ differs by $\sim 1.1\pi$ and is manifested as a phase displacement/shift of $\Delta Q(K_C,\varphi_{CEP})$. Constant $K_C = 300$, 450, and 600 eV cross-sections along the $\Delta Q(K_C,\varphi_{CEP})$ surface shown in Figure 5.14a exemplify this phase shift. Over this energy range, only Q_0 and φ_0 are changing while the amplitude $A(K_C)$ remains nearly constant. As K_C continues to increase beyond 600 eV, $\varphi_0(K_C)$ is approximately constant while $A(K_C)$ decreases to zero as the spectral components of the kinetic energy distribution vanish. The specific values of φ_0, Q_0, and $A(K_C)$ are intricately coupled to the exact position of K_C with respect to dynamical δ_1 and δ_2. Evidently, it is possible to tailor K_C to arrive an optimal value of $\zeta(K_C) = 60\%$ at $K_C = 600$ eV. Through knowledge of the Q waveform and its functional dependence on the CEP, a novel

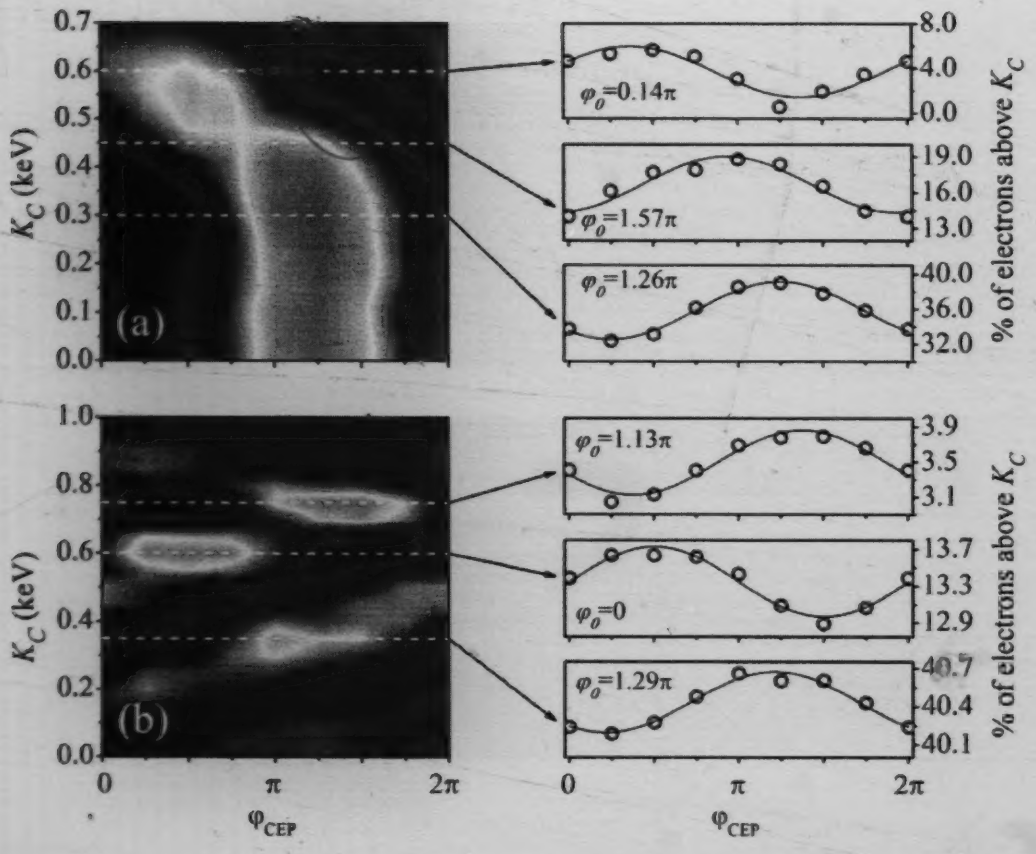

Figure 5.14. $\Delta Q(K_C, \varphi_{CEP})$ surface plots illustrating the electron count as a function of both K_C and φ_{CEP} for (a) $\tau_p = 5$ fs and (b) $\tau_p = 12$ fs. Constant K_C cross-sections along surfaces are shown for both $\tau_p = 5$ fs and 12 fs, indicating that $Q(K_C, \varphi_{CEP})$ can be tailored to yield either 'sine-like' or 'cosine-like' waveforms.

absolute 'CEP-meter' can be developed for titanium-sapphire oscillators: a feat that has yet to be accomplished.

It is also important to illustrate that a phase-sensitive $\Delta Q(K_C, \varphi_{CEP})$ surface can be achieved for even longer duration optical pulses of $\tau_p = 12$ fs as shown in Figure 5.14b. Here, five distinct regions, corresponding to the number the optical cycles ($n \sim 5$), are evident in the phase-sensitive map. Examination of constant $K_C = 350$, 600, and 750 eV cross-sections along the $\Delta Q(K_C, \varphi_{CEP})$ surface demonstrate that $Q(K_C, \varphi_{CEP})$ can be either 'sine-like' or 'cosine-like'; however, the optimal $\zeta(K_C)$ is reduced to 10% for $K_C = 750$ eV. This important result also indicates potential for absolute CEP measurement, and furthermore, relaxes the restriction $\tau_p \sim T_0$.

5.3 Summary

In this chapter, two physical process surrounding SP electron acceleration have been explored. First, a novel method for electron beam slicing has been proposed and theoretically investigated. Since the generation of the SP wave relies on ultrashort optical pulses, a large portion of an incident electron beam can be temporally gated with a precision limited only by the ponderomotive interaction, and results in an electron pulse having a temporal duration similar to that of the optical pulse. The sliced electron pulse is highly directional and investigation of its spatial distribution reveals a large degree of microbunching. It is expected that implementation of a shorter wavelength for excitation of the SP wave, even shorter duration electron bunches can be created, which is essential for increasing the resolution of time-resolved experiments utilizing ultrashort electron

pulses. Angle-resolved energy spectra reveal discrete energy bands, illustrating that the deflected electron energy and angle are interrelated, thus affording a method for isolating the electron bunches. Furthermore, it is shown that the SP gating mechanism can be utilized for temporal characterization of ultrashort electron bunches below 100 fs. Since the scheme relies on all-optical ponderomotive acceleration, it is limited only by the characteristics of the laser pulse used to excite the SP wave and offers the hope of breaking the '10 fs' barrier.

Second, it is demonstrated that SP electron acceleration can be coherently controlled through the carrier-envelope phase (CEP) of the excitation optical pulse. It is shown through model calculations that the kinetic energy gain experienced by an electron in the electric field of the SP wave depends intrinsically on the CEP. Analysis indicates that the physical origin of the CEP-sensitivity arises from an electron's ponderomotive interaction with the oscillating electromagnetic field of the SP wave. Furthermore, selection of a particular photoelectron energy range allows the nature of the variation of electron count with the CEP to be specifically tailored, even for pulses as long as 12 fs (5 optical cycles). The demonstration of CEP-control over the acceleration process represents a significant advance in the understanding of electric field driven processes in solid-state systems. It is also vital for envisioning new CEP measurement devices for low-energy titanium-sapphire laser oscillator systems.

5.4 References

[1] B. J. Siwick, A. A. Green, C. T. Hebeisen, and R. J. D. Miller, "Characterization of ultrashort electron pulses by electron-laser pulse cross correlation," *Optics Letters*, vol. **30**, pp. 1057-1059 (2005).

[2] A. R. Melnyk and J. R. Harrison, "Resonant Excitation of Plasmons in Thin Films by Electromagnetic Waves," *Physical Review Letters*, vol. **21**, pp. 85-88 (1968).

[3] P. Agostini, J. Kupersztych, L. A. Lompre, G. Petite, and F. Yergeau, "Direct evidence of ponderomotive effects via laser pulse duration in above-threshold ionization," *Physical Review A*, vol. **36**, pp. 4111-4114 (1987).

[4] M. van Exter and A. Lagendijk, "Ultrashort Surface-Plasmon and Phonon Dynamics," *Physical Review Letters*, vol. **60**, pp. 49-52 (1988).

[5] Ch. Lemell, X.-M. Tong, F. Krausz, and J. Burgdörfer, "Electron Emission from Metal Surfaces by Ultrashort Pulses: Determination of the Carrier-Envelope Phase," *Physical Review Letters*, vol. **90**, pp. 076403 (2003).

[6] A. Apolonski, P. Dombi, G. G. Paulus, M. Kakehata, R. Holzwarth, Th. Udem, Ch. Lemell, K. Torizuka, J. Burgdörfer, T. W. Hänsch, and F. Krausz, "Observation of Light-Phase-Sensitive Photoemission from a Metal," *Physical Review Letters*, vol. **92**, pp. 073902 (2004).

[7] S. E. Irvine, A. Dechant, and A. Y. Elezzabi, "Generation of 0.4-keV Femtosecond Electron Pulses using Impulsively Excited Surface Plasmons," *Physical Review Letters*, vol. **93**, pp. 184801 (2004).

[8] T. Tsang, T. Srinivasan-Rao, and J. Fischer, "Surface-plasmon field-enhanced multiphoton photoelectric emission from metal films," *Physical Review B*, vol. **43**, pp. 8870-8878 (1991).

[9] F. F Chen, *Introduction to Plasma Physics and Controlled Fusion*, 2nd Edition (Plenum, New York, 1984).

[10] S. E. Irvine and A. Y. Elezzabi, "Surface-plasmon-based electron acceleration," *Physical Review A*, vol. **73**, pp. 013815 (2006).

[11] J. Kupersztych, P. Monchicourt, and M. Raynaud, "Ponderomotive Acceleration of Photoelectrons in Surface-Plasmon-Assisted Multiphoton Photoelectric Emission," *Physical Review Letters*, vol. **86**, pp. 5180-5183 (2001).

Chapter 6.0

Conclusion and Summary

Research described in this thesis represents several important steps along a path towards understanding surface plasmon (SP) electron dynamics and the fundamental light-matter interactions that lay beneath this unique and fascinating form of particular acceleration. Initial evidence of this came with the demonstration of 0.4 keV electron pulse generation using only a simple titanium-sapphire laser oscillator system [1]. In contrast to preliminary work, which indicated that such energetic electron pulse generation using SP waves required high-energy laser pulses (~1 mJ), the research presented here indicates that the same level acceleration can be achieved using only 1.5 nJ pulses. Furthermore, the measured electron energy spectra were much narrower than those produced in preliminary studies. The advantages of such an achievement are clear: superior quality energetic electron pulses can be produced with fewer resources (i.e. simpler laser system) and at a higher repetition rate necessary for sensitive experiments. Such an achievement is extremely important for the development of sensitive time-resolved electron diffraction studies.

Experiments in the second stage of this project involved high-energy pulses (~ 0.5 mJ) from a multi-pass titanium-sapphire laser-amplifier system. Here it was established that electrons with energies extending beyond 2 keV could be produced via SP waves on both silver and gold metal films [2,3]. Further measurements of the angular spectra indicate that the photo-acceleration is highly directional. It was also confirmed that tunneling effects, rather than multiphoton electron emission, dominate the SP enhanced photoemission process at laser intensities above 2 GW/cm^2 in both metals. This particular fact is both intriguing and exciting: laser intensities of only a few GW/cm^2 are necessary to access the transition to this tunneling regime, as opposed to the ~100 TW/cm^2 required

for the analogous process in atomic ionization. This could potential open a doorway for studying high-field processes in solid-state systems.

To compliment the experimental discoveries, a novel computer model was developed, which was based on fundamental physical descriptions of electron emission and light-wave dynamics. Electron emission was accounted for through empirical multiphoton statistics, while Maxwell's equations were solved numerically to account for the electrodynamics [4]. Combining these two models into a comprehensive simulation aided in the interpretation of the underlying physical phenomena, provided a unique visualization of the acceleration process, and revealed novel bunching dynamics of the photo-accelerated electron pulse. The model also produced data that was in excellent agreement with the experimental results, which indicated that surface roughness of the metallic films plays an important role in the enhancement of the electric field.

6.1 Future directions

While the principal steps outlined in this thesis have developed the understanding of SP electron acceleration, they have also laid the foundation for many future research endeavors and have unlocked entirely new fields. The following section briefly describes the future research directions for SP electron acceleration.

Light-wave synthesis

Perhaps one of the most important outcomes of this work is the carrier-envelope phase analysis of the SP electron acceleration process presented in Chapter 5. In cases where the duration of an optical pulse becomes comparable to a single oscillation of the light-

wave (2.7 fs at 800 nm), the carrier-envelope phase parameter becomes important for optically driven processes. Again, the carrier-envelope phase is a measure of the timing between the peak of the envelope of the pulse and the maximum of the underlying light-oscillation. By shifting the underlying waveform (i.e. changing the carrier-envelope phase), however, the outcome of a phase sensitive physical process can be altered. Through the unique model presented in this thesis [4], it was shown that the carrier-envelope phase of a light wave could be used to manipulate the SP electron acceleration process [5]. Such control represents an entirely new and exciting research field of coherently controlled laser acceleration at metallic surfaces, and further developments would lead to significant advancement in areas such as particle acceleration, high-harmonic generation from solids, chemical/biological diagnostics, and material science.

Particularly, the model results presented in Chapter 5 indicated that pronounced cut-offs appear within the kinetic energy spectra of the electrons. It was revealed that these cut-offs are due to the very nature of the ponderomotive interaction and that the number of cut-offs corresponds to the number of cycles in the optical pulse. As the carrier-envelope phase value is changed, the positions of the cut-offs shift within the energy spectra. Future experiments must focus on finding these cut-offs in the energy spectra and determining if they behave as the model predicts. Once this is realized, a number of intriguing experiments would follow. Perhaps the most important would be the development of carrier-envelope phase measurement device, or a 'phase-meter' as depicted in Figure 6.1. By selecting an energy range within the electron spectra, the photocurrent varies directly with the carrier-envelope phase, and thus, affords a method for absolute characterization of the electric field waveform. Since SP electron

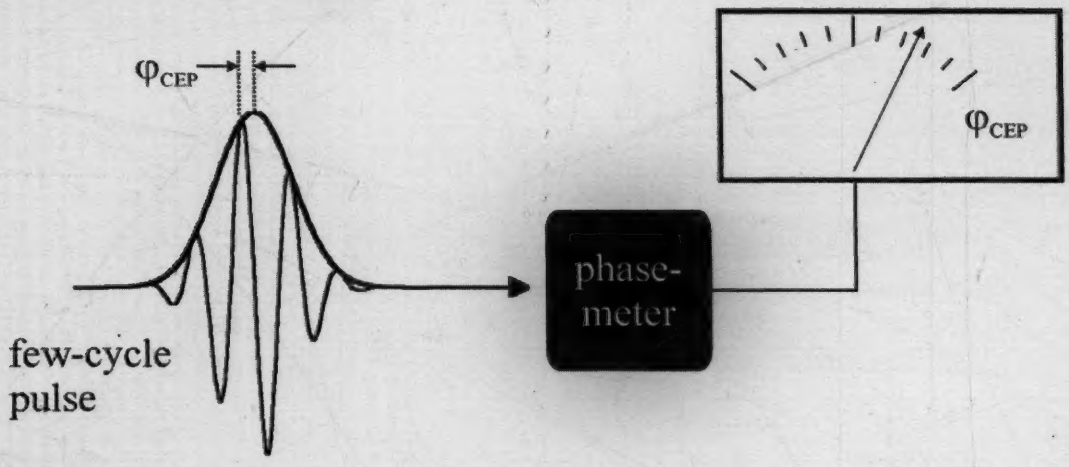

Figure 6.1. Depiction of a phase-meter: a device capable of determining the absolute carrier-envelope phase of a few-cycle laser pulse.

acceleration can be achieved with low-energy pulses (~1 nJ), a novel absolute phase-meter can be developed for titanium-sapphire oscillators: a feat that has yet to be accomplished. This is extremely important for calibration of few-cycle laser sources, since, at the state-of-the-art, only the pulse-to-pulse (i.e. relative) carrier-envelope phase shift can be measured. Indeed, such a calibration is necessary for interpreting phase-sensitive experiments that employ titanium-sapphire oscillators.

Other possible experiments include: investigating the angular distribution of the electrons in the few-cycle regime, determining the influence of the particular material upon which the SP is confined to, the morphology of the metallic surface (i.e. surface roughness effects), bimetallic interfaces (where the plasma frequency and plasmon damping are governed by the metal-metal contact potential), and implementation of a phase-stabilize amplifier to access higher intensities (above a few GW/cm^2). Indeed, these experiments are invaluable, as control over basic processes through the electric

field of a light-wave ultimately leads to new knowledge of fundamental light-matter interaction phenomena.

Nano-engineering of metallic structures

The prospect of generating energetic femtosecond electron bunches using a simple laser oscillator has, of its own merit, opened a doorway to many other avenues of research including new schemes for generating localized electrons packets using laser radiation. In particular, electron emission from nano-engineered structures offers nearly endless possibilities. As already demonstrated in this work, electron emission is highly dependent upon the surface morphology of the metal film under investigation. The evolution of the metallic surface during its growth process is highly dependent on the particular conditions in which the film is grown [6]. Data presented in Chapter 4, also shown in Figure 6.2a and 6.2b, shows two silver surfaces deposited by different vacuum metallization techniques. Clearly, the surface shown in Figure 6.2a contains a large number of nanometric protrusions (<50 nm). Therefore, complete characterization of the electron acceleration with respect to the deposition parameters would yield the optimal surface morphology for the ponderomotive interaction. Along similar lines, nano-engineering surfaces can also be investigated. Figures 6.2c-6.2d illustrate fabricated nano-particles via pulsed laser deposition [7,8] and periodic nano-structures [9] that may exhibit enhanced photoemission and acceleration to yield a larger number of higher energy electrons. Furthermore, such nano-assemblies can also be studied with respect to composition, in which various types or combinations of different metals and dielectrics can be explored.

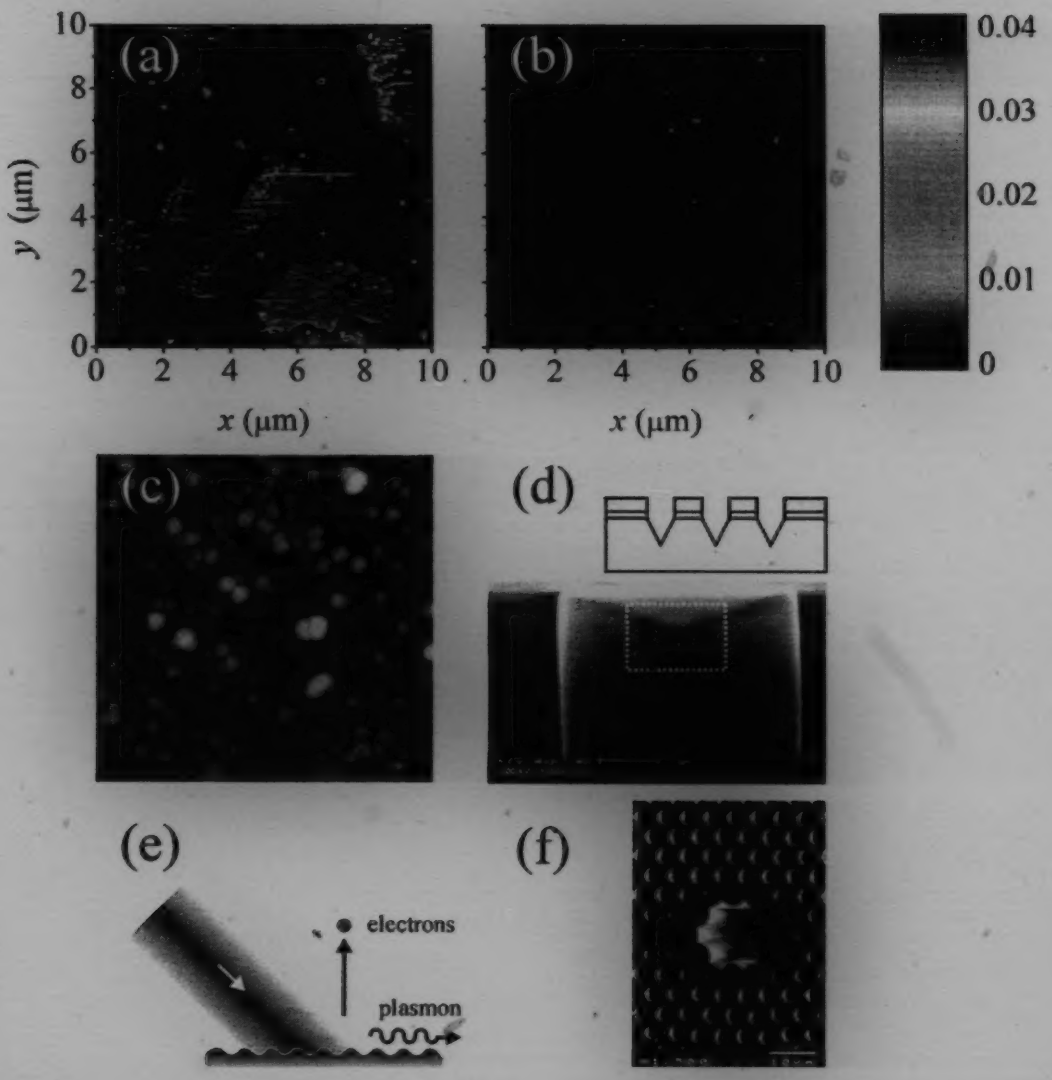

Figure 6.2. The first two panels illustrate atomic force microscope measurements, repeated from Chapter 4, of two silver surfaces that have been deposited using two different methods of metallization: (a) magnetron sputtering and (b) ion-beam sputtering. The next four panels illustrate other types of nano-structures that hold promise for research of electron acceleration and electron pulse dynamics. Panel (c) shows nano-particles created by pulsed laser deposition. Periodic nano-pyramids, depicted in (d), can be created through anisotropic etching of a silicon surface. Other geometries that can be investigated include the (e) grating method for launching surface plasmons as well as (f) the holey fiber. In the latter case, the fiber tip can be either coated with a metal film or impregnated with metallic nano-particles.

Alternate SP coupling geometries offer unexplored potential for creating ultrashort electron bursts. The grating configuration shown in Figure 6.2e, which provides the necessary momentum matching for plasmon coupling, does not introduce material dispersion, and hence, ensures that the optical pulse duration remains at its minimum transform-limited value. This results in a larger electric field for the ponderomotive interaction as well as a shorter electron pulse. The periodicity of the grating can also be varied to investigate the coupling mechanism and maximize the enhancement of the electric field near the surface.

The advent of photonic band-gap fibers, or holey fibers (Figure 6.2f), has provided access to fundamental non-linear processes using low-energy titanium-sapphire oscillators. Using these holey fibers, SP dynamics can be studied via the electron emission and acceleration processes in the presence of the nonlinear radiation. The SP waves can be launched on a metal film that has been deposited on or near the fiber, or on nano-particles that have been injected directly into the fiber core.

Nano-accelerators

Alternatively, the entire SP coupling and electron emission geometry can be completely engineered into a novel electron nano-accelerator as shown in Figure 6.3. Here, a metal-coated fiber tip can be used to simultaneously focus the laser radiation and launch SP waves on the nanometer-sized tip. The tip of the fiber can be manufactured using a HF acid bath [10] to produce a sharp nanometer-sized point (< 100 nm), which is subsequently coated with a metal film. When an ultrashort pulse from a femtosecond laser travels down the fiber core, it is focused by the conical geometry of the fiber tip and

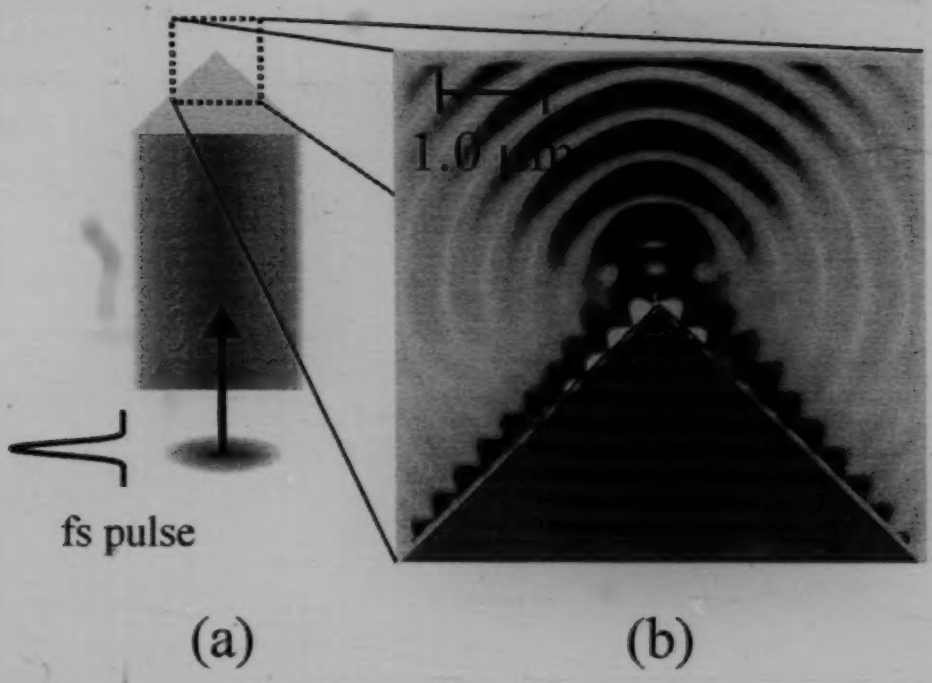

Figure 6.3. (a) Geometry for a nanometric electron accelerator, comprised of a glass optical fiber, which is tapered and metal coated at one end. (b) Depiction of the electric field distribution near the tip of such a device.

is coupled to a SP oscillation. The plasmon would then travel along the remainder of the tip to the vertex, where its electric field would be greatly enhanced. The end result is the production of extremely localized ultrashort electron bursts that would be valuable for probing nanometer structures with femtosecond resolution. These electron bursts would also be useful for other practical applications including nanolithography and directional electron nano-injectors.

Optically Engineered Plasmons for Electron Acceleration

Since the acceleration is of ponderomotive origin, both the spatial profile and polarization of the optical beam will have an effect on the electron dynamics. Numerous techniques

for generating unusual polarization states or optical vortices/lattices can be implemented with the end goal of controlling electrons with light waves. For example, early simulations results, shown in Figure 6.4, indicate that coupling of a TEM_{01} Gaussian mode can lead to an amplitude modulated plasmon wave. Since the launching of the plasmon occurs at an angle (~41° for silver), different sections of the incident beam arrive at different times. In the case of a TEM_{01} mode, the portion of the beam that arrives later has a polarization that is equal, but opposite to the other half that has already coupled to a plasmon mode, thus canceling the SP oscillation. In essence, the plasmon is 'turned on' and then 'turned off' a short time later. This demonstrates that a non-uniformly polarized beam can be used to modulate a SP on a timescale shorter that the plasmon lifetime, and ultimately, create even shorter duration electron pulses. Such a TEM_{01} mode can be easily created from the fundamental TEM_{00} mode of a laser system using a combination optical waveplates.

An example of a beam having a spatially distributed polarization is the Bessel-Gauss mode [11]. Such a mode has a donut shape (i.e. zero field in the centre) and consists of a radially varying polarization. When focused, the electric fields add to yield a longitudinal electric field (illustrated in Figure 6.4) that can be used for directional electron acceleration. Ideally, such research would be initially investigated through numerical simulation and later verified by experiments.

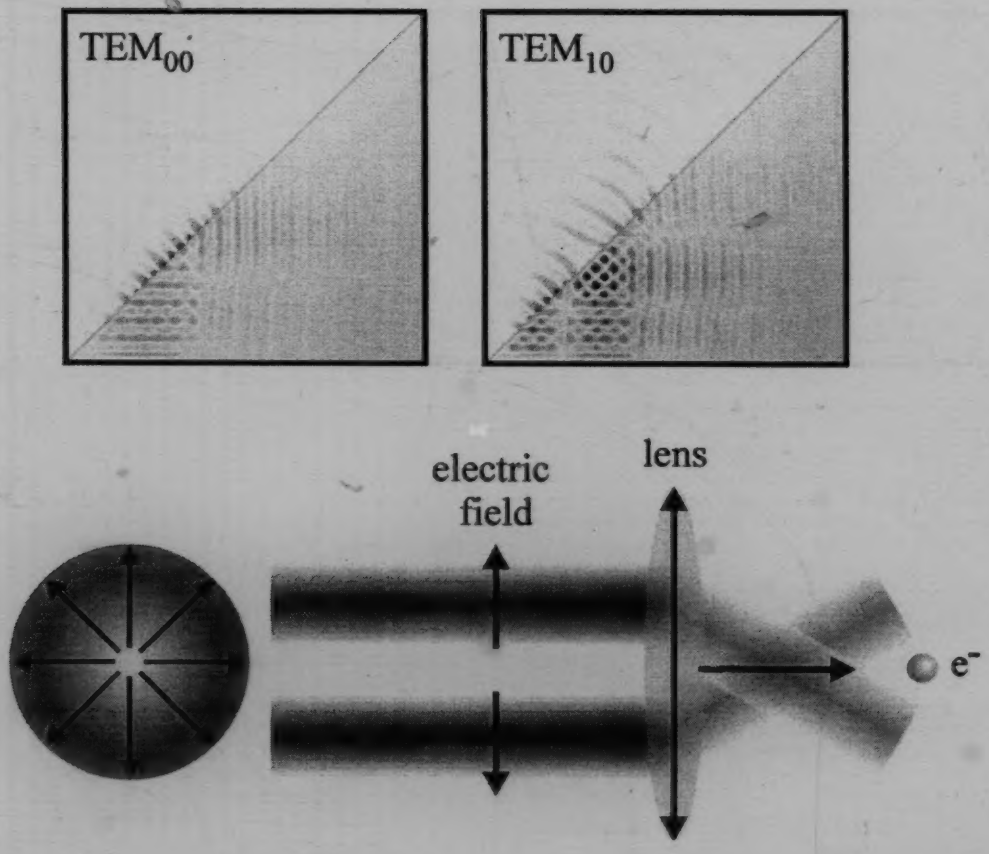

Figure 6.4. (top) Finite-difference time-domain simulation of a TEM_{00} and TEM_{10} optical pulses coupling to surfaces plasmons. In the case of the TEM_{10} mode, an amplitude modulated wave results, indicating that a spatially engineered light-wave can be implemented to switch the plasmon on a time-scale below its lifetime. (bottom) Depiction of a Bessel-Gauss mode, which, when focused, provides a longitudinal electric field that can be used to accelerate or steer electrons.

Physics of plasmon-enhanced electron emission

The emission process itself can be used as a window into the underlying physics of laser-matter interaction at metallic surfaces. Specifically, the influence of the SP on electron emission can be studied. It is known that by coupling to SP waves, the photoelectron yield can be increased by over 3 orders of magnitude [12]. Obviously, the presence of the SP mode has a drastic effect on electron emission; however, its exact influence on the quantum efficiency is not known. Is it the plasmon itself that contributes to the electron emission, or is it simply the fact that the local photon density inside the metal is increased during the launching of the plasmon?

A unique experiment that may answer this question can be implement and is described as follows. In the context of photo-ionization of molecules, it is known that two regimes exist for photoemission: multiphoton and tunnel or Keldysh ionization. Access to either can be gained by controlling the intensity of the light wave used to invoke the process. Recent work [13], including the work of this thesis, has shown that an analogous effect occurs for metal surfaces. By tracking the photocurrent generation as a function of pump intensity, it was shown [3] that for intensities above 2 GW/cm^2, the order of the photoemission process is reduced to a value lower than its normal multiphoton one (n=3 for silver and n=4 for gold at 800 nm). In the multiphoton case, electron emission is essentially an intensity driven process. However, for field emission, electron generation is dominated by tunneling effects, which constitute an electric field driven process. If the

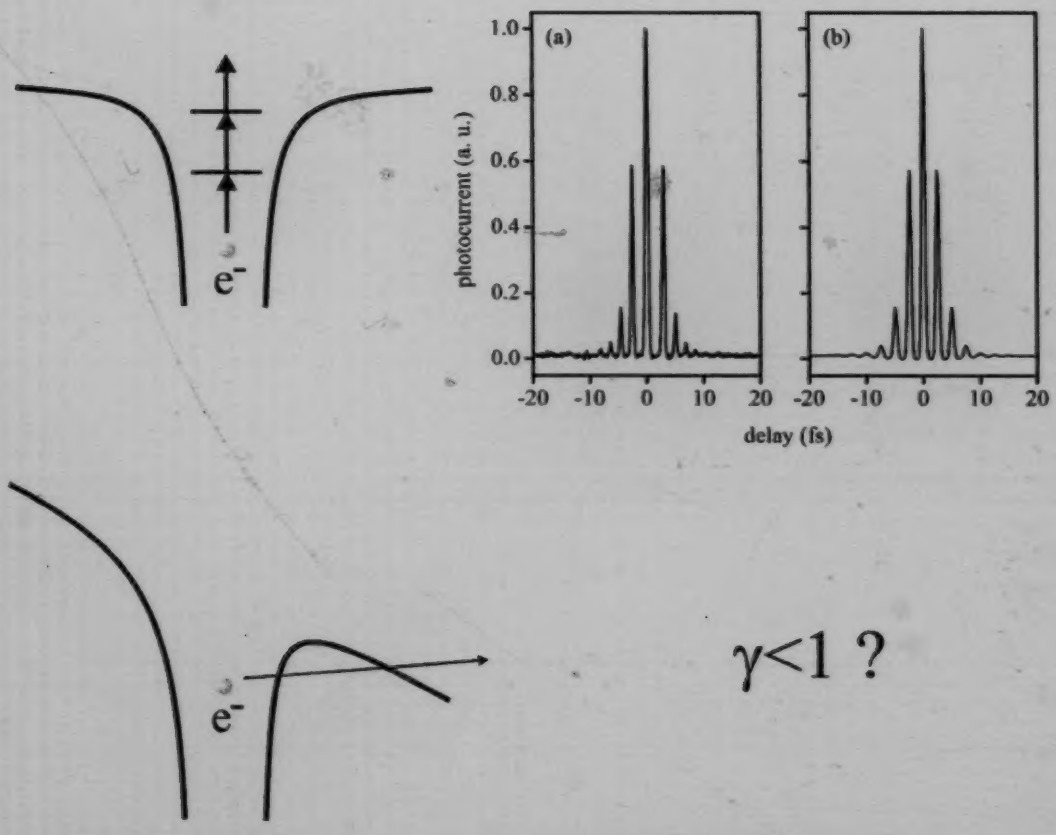

Figure 6.5. (top) Experimental data acquired using a titanium-sapphire oscillator in the multiphoton electron emission regime ($\gamma > 1$). The autocorrelation trace indicated no broadening due to the finite lifetime of the plasmon. In the case of Keldysh emission (bottom), electron emission would be correlated with the presence of the electric field of the plasmon. Therefore, it is expected that the autocorrelation trace would be significantly broadened as the electron emission is now driven by the electric field, which persists for the entire duration of the plasmon lifetime.

electron emission is due to the electric field of the plasmon wave, then it should persist for the entire duration of the plasmon wave. Therefore, an autocorrelation experiment can be devised to ascertain this emission lifetime.

Experimental results, shown in Chapter 4, have already been obtained for the multiphoton regime using 5 fs laser pulses. The autocorrelation function shown in Figure 6.5 (top) exhibits no measurable broadening due to an intermediate process, i.e. electron emission is instantaneous. However, in the tunnel regime, it is expected that such an autocorrelation would be significantly broadened by the lifetime (48 fs [14]) of the SP wave, as the emission would now be correlated with the presence of a SP electric field. Therefore, as the intensity of the two beams is increased beyond the transition from the multiphoton to the tunnel regime (using a phase-stabilized amplifier system), the autocorrelation function should broaden significantly. A direct comparison between the two would yield the lifetime of the SP wave.

Time-resolved electron diffraction

The electron pulse generation system developed will serve as an ideal candidate for time-resolved electron diffraction. As depicted in Figure 6.6, electron pulses can be used to accumulate single frames of an entire 'motion picture' of femtosecond phenomena. However, a contemporary issue for time-resolved electron diffraction studies is the generation and measurement of sub-100 fs electron pulses. Typical electrostatic systems can, at best, generate electron pulses as short as 100 fs, and as a result, alternative methods are sought after which are not limited by large interaction distances (> 1 cm) and deleterious space-charge effects. Realizing this, a novel technique of electron beam

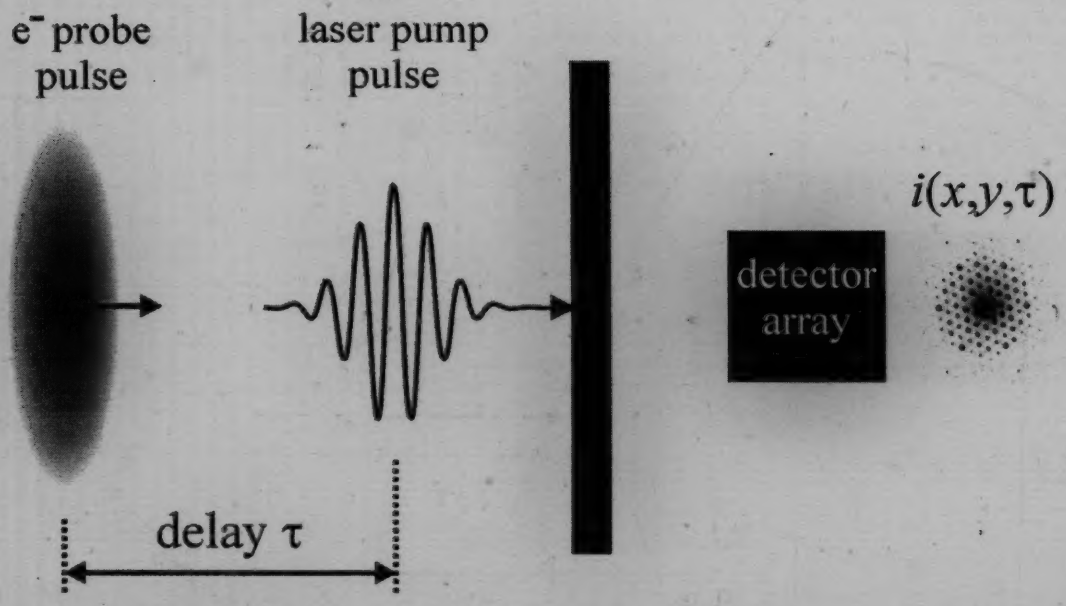

Figure 6.6. Conceptual illustration of a time-resolved electron diffraction experiment. A laser pump pulse is used to excite a system, and a short time later, an electron pulse probes the excited system. When the delay between the two pulses is varied, a 'motion picture' of the physical process can be created from the time-varying diffraction pattern and used to determine ultrashort dynamics of the physical process in question.

slicing for the generation or characterization of sub-100 fs electron packets has been proposed [15] and described in Chapter 5. Since the scheme relies on all-optical ponderomotive acceleration, it is limited only by the characteristics of the laser pulse used to excite the SP wave. Using the model developed in this thesis (Chapter 3), it is demonstrated that electron packets having durations less than 100 fs can be generated. Furthermore, it is shown that the temporal profile of a pre-existing electron pulse can be characterized through optical-electron cross correlation. These simulation results are very promising indeed and would represent a both challenging and intriguing experimental venture.

6.2 References

[1] S. E. Irvine, A. Dechant, and A. Y. Elezzabi, "Generation of 0.4 keV Femtosecond Electron Pulses using Impulsively Excited Surface Plasmons," *Physical Review Letters*, vol. **93**, pp. 184801 (2004).

[2] S. E. Irvine and A. Y. Elezzabi, "Ultracompact 180° magnetic spectrometer for intermediate energy electron measurement," *Measurement Science and Technology*, vol. **17**, pp. 2455-2460 (2006).

[3] S. E. Irvine and A. Y. Elezzabi, "Ponderomotively Accelerated 2 keV Femtosecond Electron Packets using Surface Plasmon Waves," *Applied Physics Letters*, vol. **86**, pp. 264102 (2005).

[4] S. E. Irvine and A. Y. Elezzabi, "Surface-plasmon-based electron acceleration," *Physical Review A*, vol. **73**, pp. 013815 (2006).

[5] S. E. Irvine, P. Dombi, Gy. Farkas, and A. Y. Elezzabi, "Influence of Carrier-Envelope Phase of Few-Cycle Pulses on Surface-Plasmon Ponderomotive Electron Interaction," *Physical Review Letters*, vol. **97**, pp. 146801 (2006).

[6] J. A. Thornton, "Influence of apparatus geometry and deposition conditions on the structure and topography of thick sputtered coatings," *Journal of Vacuum Science and Technology*, vol. **11**, pp.666-670 (1974).

[7] Z. Pászti, G. Petõ, Z. E. Horváth, and A. Karacs, "Laser ablation induced formation of nanoparticles and nanocrystal networks," *Applied Surface Science*, vol. **168**, pp. 114-117 (2000).

[8] J. P. Barnes, A. K. Petford-Long, R. C. Doole, R. Serna, J. Gonzalo, A. Suárez-García, C. N. Afonso, and D. Hole "Structural studies of Ag nanocrystals

embedded in amorphous Al_2O_3 grown by pulsed laser deposition," *Nanotechnology*, vol. **13**, pp. 465-470 (2002).

[9] D. W. Kim, J. T. Ok, S. S. Choi, C. K. Chun, J. W. Kim, and J. H. Boo, "Analysis of the aperture formation mechanism in the fabrication process of nano-aperture arrays," *Microelectronic Engineering*, vol. **73-74**, pp. 656–661 (2004).

[10] B. A. F. Puygraniera and P. Dawson, "Chemical etching of optical fibre tips-experiment and model," *Ultramicroscopy*, vol. **85**, pp. 235-248 (2000).

[11] F. Gori, G. Guattari, and C. Padovani, "Bessel-Gauss beams," *Optics Communications*, vol. **64**, pp. 491-495 (1987).

[12] T. Tsang, T. Srinivasan-Rao, and J. Fischer, "Surface-plasmon-enhanced multiphoton photoelectric emission from metal films," *Physical Review B* **43**, 8870-8878 (1990).

[13] C. Tóth, G. Farkas, and K. L. Vodopyanov, "Laser-Induced Electron Emission from an Au Surface Irradiated by Single Picosecond Pulses at λ=2.94 μm. The Intermediate Region Between Multiphoton and Tunneling Effects," *Applied Physics B*, vol. **53**, pp. 221-225 (1991).

[14] M. van Exter and A. Lagendijk, "Ultrashort Surface-Plasmon and Phonon Dynamics," *Physical Review Letters*, vol. **60**, pp. 49-52 (1988).

[15] S. E. Irvine and A. Y. Elezzabi, "Femtosecond electron pulse gating using surface plasmons," *Optics Express*, vol. **14**, pp. 4115-4127 (2006).

Appendix A

Magnetic Spectrometer Design

A.1 Introduction[1]

Particle energy analysis lies at the heart of numerous diagnostic and spectroscopic techniques in areas such as laser-plasma interaction, electron diffraction, and high-energy physics. Surface science alone relies on a host of methods that use electrons to probe matter and gain information regarding physical structure, composition, and chemistry. Examples include, but are not limited to: x-ray photoelectron spectroscopy, low-energy electron diffraction, and Auger spectroscopy [1]. Generally, the energies of electrons measured in these spectroscopic methods lies below the keV range, and therefore, standard electrostatic [2,3] or time-of-flight [4,5] energy measurement techniques can be applied. Conversely, extremely high-intensity laser-matter interactions are capable of generating electrons with energies extending up to hundreds of keV. In such high-energy experiments, the favored choice for spectrometer design relies on large static magnetic fields to spatially disperse the electrons according to their velocities [6-8]. Despite the fact that much effort has been devoted to spectrometer design and the quantification of charged particle energies, designs for the measurement of electrons having energies in the intermediate range from 1 to 50 keV are required. While a number of alternative electrostatic spectrometer designs have been proposed [9-11] or demonstrated for measuring electrons with energies below ~50 keV, they suffer from disadvantages including complex/large design and/or high-voltage requirement.

To facilitate electron energy measurement in the intermediate range from 0-50 keV, a novel 180° magnetic electron spectrometer is designed and characterized [12]. The prototype consists of an ultracompact custom-built aluminum vacuum chamber (<20

---

[1] A version of this chapter has been published: S. E. Irvine and A. Y. Elezzabi, Measurement Science and Technology, **17**, 2455-2460, 2006. Copywrite (2006) IOP Publishing Ltd.

cm^3), two solenoid coils for the generation of a variable magnetic field required for electron dispersion, and an integrated multiplier for electron detection. An interesting feature of this design is that the magnetic coils are arranged external to the chamber, thus alleviating high-vacuum compatibility and heat dissipation issues. In addition, the ultra-compact size allows for a reduced manufacturing cost, space requirement, and vacuum pump-down time. Test particle code is designed and implemented to illustrate the device operation and determine its resolution.

A.2 Device Schematic and Operational Principles

A schematic diagram of the spectrometer is shown in Figure A.1 and A.2. Overall, the device is comprised of three main components: two magnetic field generation coils and a central chamber that is placed between the coils (Figure A.1). The two coils, machine-wound using 18-gauge insulated copper wire and each having a total electrical resistance of 4.7 Ω, are placed in direct contact with the central aluminium chamber and generate a magnetic field, $B_z(x,y)$, that spatially disperses incoming electrons according to their energy. In contrast to most other high-energy designs [6,7], which use a static magnetic field to spatially disperse electrons onto an extended multichannel detector, the operational principle of this device is to curve electrons through a *fixed* arc using an externally applied magnetic field [8]. For a given magnetic field, electrons of a particular energy will follow a curved trajectory, and thus, by varying the current, I, within the coils (and hence magnetic field strength), the energy spectrum of an electron source can be determined for a fixed location of a single detector. As shown in Figure A.2, the main central chamber is constructed of a single solid piece of copper-plated aluminium that

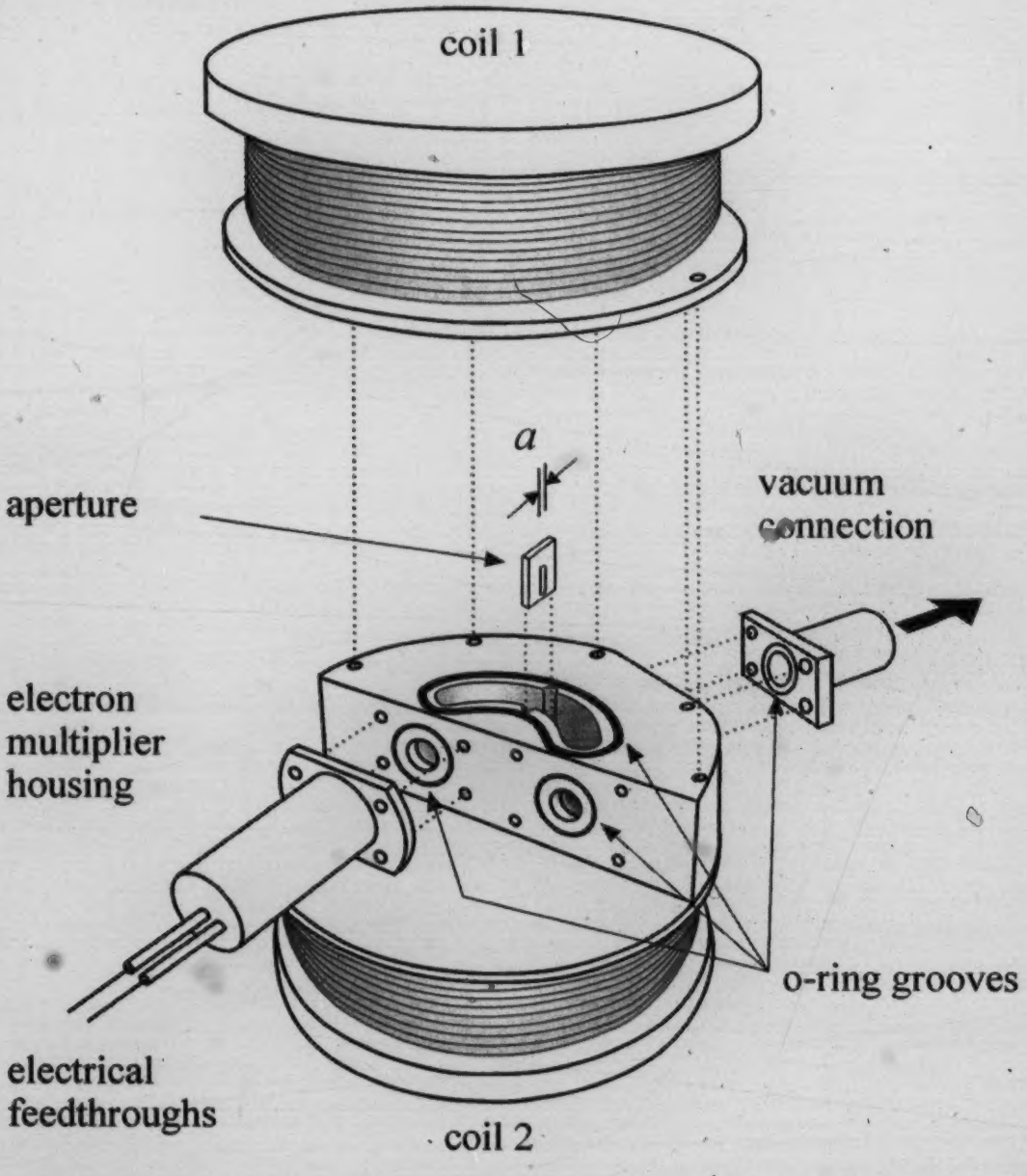

Figure A.1. A variable field magnetic spectrometer, comprising a central fixed-arc chamber through which electrons flow and two magnetic field generation coils required to disperse the electrons according to their velocity. Electron detection is achieved using an electron multiplier that is situated in its own housing containing the necessary electrical feedthroughs. Evacuation of the chamber occurs through a vacuum port, also attached directly to the central chamber and located at the rear. To increase the resolution of the device, an aperture, of width a, is placed within chamber in the direct path of the electrons.

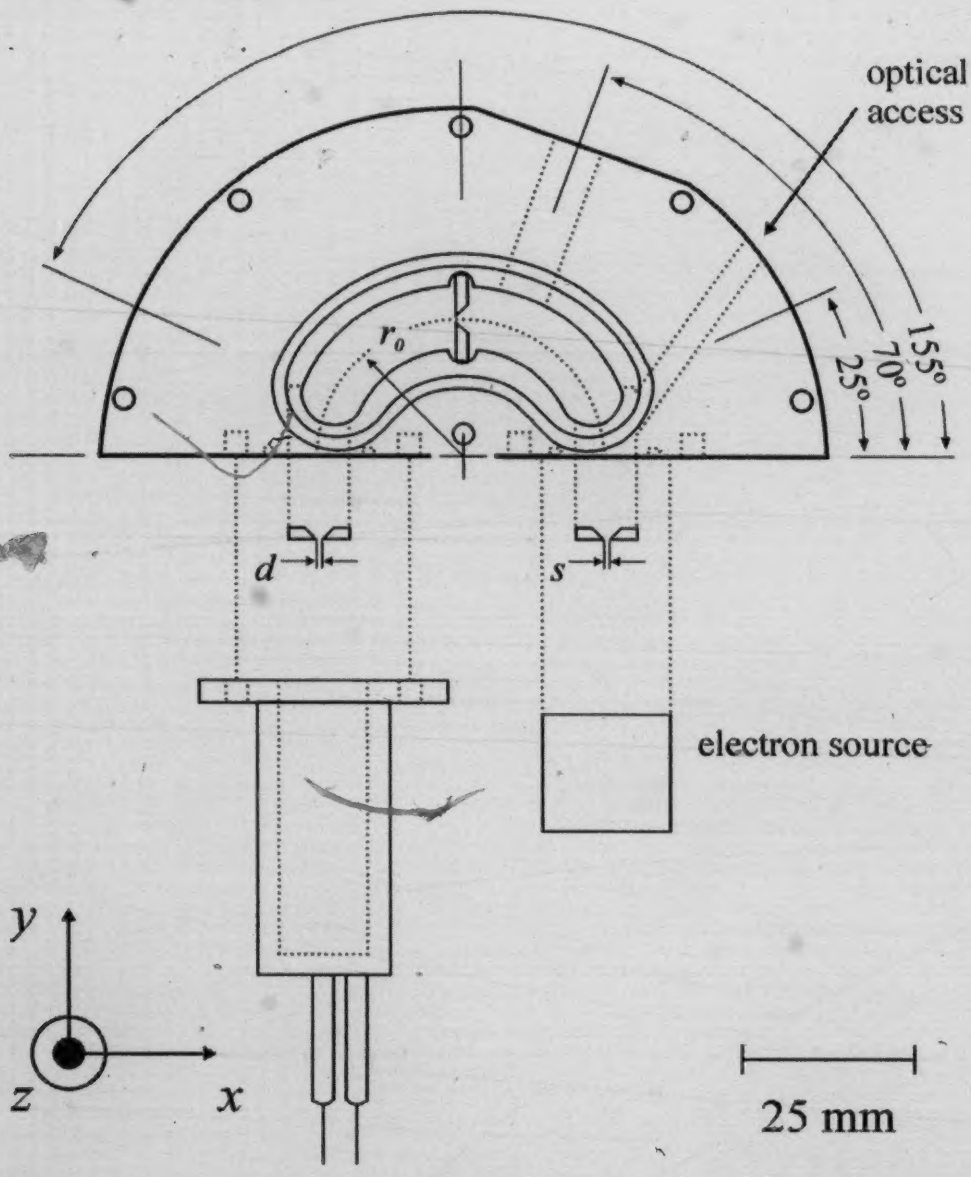

Figure A.2. Schematic diagram of the central chamber and various adjoining components. The main chamber consists of a single fixed-arc of radius of curvature r_0, which is evacuated through a port located at 70°. Electrons, generated at the source, will follow a trajectory through the chamber that depends on the current, I, in the coils and their initial velocity. Depending on the conditions of the electron source, an optical access port is also included in the design to allow for front-illumination of samples. To restrict the number of possible electron trajectories and increase the spectrometer resolution, apertures of widths of s, d, and a are placed at the source, detector, and mid-chamber, respectively.

serves both as the measurement device and vacuum chamber. A central 180° semicircle of a fixed radius of curvature, r_0, inner radius r_1, and outer radius r_2, through which electrons will travel, is constructed through standard milling. First, a single arc is milled from 25° to 155° at a radius of $r_0 = 19.1$ mm (0.75"), with $r_1 = 23.1$ mm and $r_2 = 15.1$ mm, while the remainder of the arc is formed by drilled straight vias located at 0° and 180°. This allows for the necessary vacuum sealing surfaces at 0° and 180° for the electron source and detector assemblies, respectively. An additional o-ring seal surrounding the central arc is created using a computer-controlled mill and forms a seal when fastened against coil 1. More importantly, however, is the fact that this fixed arc restricts electron flow, from the source to the detector, to a particular energy, E, and energy spread, ΔE. A general electron source, complete with its own housing, is depicted in Figure A.2 and connects directly to the system. For optical-matter interaction (e.g. photoelectron spectroscopy, laser-matter interaction), the electron source housing would include a window for optical throughput. A channel can also be included for cases that require front-side excitation of a sample surface. Depending on the specific application, the angle of this channel can be altered to suit the particular experiment. High-vacuum pumping (10^{-5}-10^{-6} Torr) of the <20 cm^3 volume is accomplished through an access port that is connected to the main chamber. All connections and fasteners are constructed from non-ferromagnetic materials to avoid irregular field distributions and fringing near edges.

Due the finite spatial extent of the source and detector, electrons having a large ΔE can reach the detector and limit the overall resolution, $\Delta E/E$, of the device. To obtain adequate precision in measurement ($\Delta E/E \leq 10\%$), thin diaphragms having

aperture widths of s, d, and a are placed at the source, detector, and within the chamber (at 90°, see Figure A.2), respectively. The addition of these diaphragms limits the number of possible electron paths and decreases the overall uncertainty in measured energy. Depending on the nature of the experiment, the size and location of the apertures can be adjusted to achieve a balance between signal amplitude and the required $\Delta E/E$.

Electron detection is achieved using a compact continuous dynode electron multiplier (Ceramax 7596m). The electron multiplier is contained within a separate aluminium housing that connects and seals directly to the central chamber. This housing contains two ceramic electrical feedthroughs required for high-voltage biasing and are sealed by vacuum compatible epoxy. A schematic diagram of the electrical biasing circuit is shown in Figure A.3. Detection of electrons necessitates that the entrance of the electron multiplier is electrically grounded, while electron multiplication required a large positive voltage gradient along the length of the device. As a result, the small electrical signal produced by incoming electrons must be decoupled from the large static potential at the electron multiplier anode using a high-voltage capacitor (3 kV, 10 nF). High-speed diodes (Philips Semiconductor, 1N4148, 4 ns switching time) are used to shunt any high-voltage transients (>0.5 V) produced in the electron multiplier circuit (e.g. arcing, power failure) and protect subsequent electronic amplifier stages required for signal detection/processing.

A.3 Results and Discussion

A.3.1 Magnetic Field Distribution and Trajectory Analysis

The particular trajectory that an electron will follow through the spectrometer depends highly on the magnetic field distribution within the chamber. Therefore, it is important to produce a smooth magnetic field, as large fluctuations can cause severe aberrations in the electron-optical system. The measured magnetic field distribution, $B_z(x,y)$, is shown in Figure A.4 for $I = 1.0$ A along an *x*-*y* plane that corresponds to the vertical centre of the spectrometer. An outline of the central arc of the main chamber is overlaid with $B_z(x,y)$ to illustrate the distribution in the region of electron flow. The peak value of $B_z = 5.6$ mT occurs at the centre of the coils and drops to minima near the top corners of the plot. Within the interaction region, the magnetic field changes by only ~10%. Two additional magnetic field distributions taken at elevations of $z = -4.0$ mm and $z = 4.0$ mm yield nearly identical field values compared to the data shown in Figure A.4.

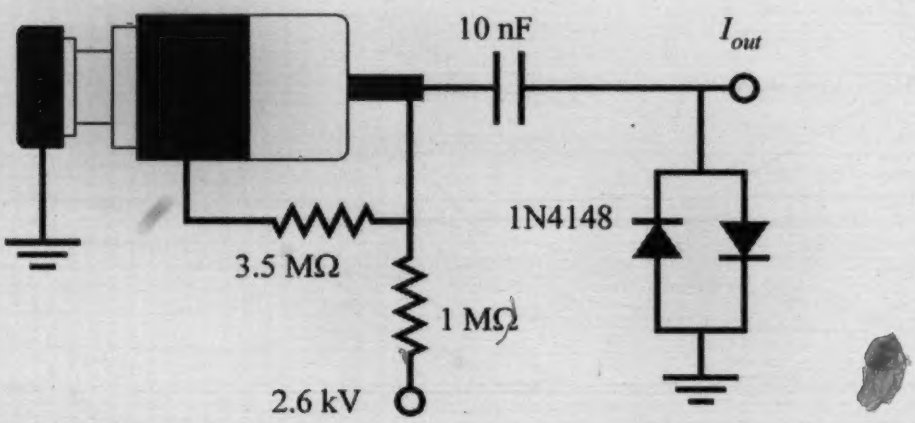

Figure A.3. Electrical schematic of the circuit used to bias the electron multiplier. Fast 1N4148 diodes are used to protect subsequent amplifier/processing electronics from high-voltage transients.

Calculation of the measured energy and resolution of the device at a given I requires knowledge of the exact trajectories of electrons that are subjected to $B_z(x,y)$. The 10% variation in the magnetic field distribution within the spectrometer precludes any analytic derivation of E and ΔE at a given current. Hence, test-particle code is designed and implemented to trace the path of a sample set of electrons that exit the source and traverse the $B_z(x,y)$. To represent all possible trajectories, 10^7 electrons are emitted from the source region. Their velocity vectors are uniformly distributed over the angular and energy ranges of $\theta = (-\alpha, \alpha)$ and $K = (0, K_0)$. For each trajectory, the Lorentz force equation

$$\frac{d\vec{v}}{dt} = \frac{q}{m_e}\vec{v} \times \vec{B} \tag{A.1}$$

is solved, where q, m_e, and $\vec{v}$ are the charge, mass, and velocity of the electron, respectively, and $\vec{B} = (0,0,B_z(x,y))$ is the magnetic field strength. The velocity and position of each electron is determined by solving equation A.1 using a fourth-order Runge-Kutta numerical method [13] in which the magnetic field values are bilinearly interpolated from the measured values shown in Figure A.4. Once the simulation is complete, the electrons are sorted into equally distributed kinetic energy bins to determine the transmitted electron energy spectra for a given current, I. Figure A.5 illustrates a sample set of electron trajectories that are emitted from the source and successfully reach the detector for $I = 1.0$ A, $s = 2$ mm, $d = 1$ mm, $a = 1$ mm, $\alpha = 60°$, and $K_0 = 50$ keV. Any electron that traces a path into the surrounding copper-plated metal chamber is reabsorbed and discarded from the simulation.

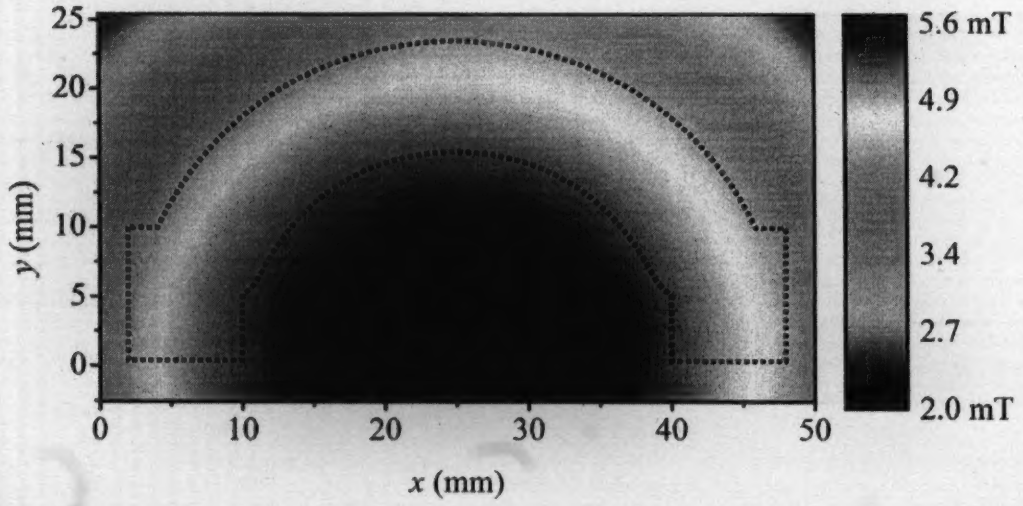

Figure A.4. Distribution of the $B_z(x, y)$ component of the magnetic field generated by the coils at a current of $I = 1.0$ A. The dotted line indicates the outline of the central arc in which electrons flow.

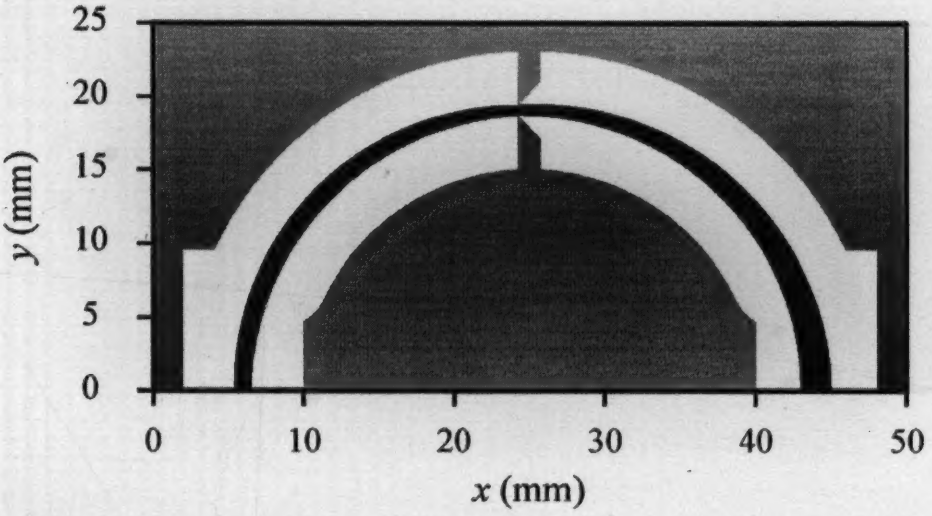

Figure A.5. Sample set of electron trajectories that are emitted from the source and successfully reach the detector for $I = 1.0$ A, $s = 2$ mm, $d = 1$ mm, and $a = 1$ mm, $\alpha = 60°$, and $K_0 = 50$ keV. Here, 0° represents the normal or positive y-direction.

The resultant transmitted energy spectrum of the sample set of electrons shown in Figure A.5 is used to determine the central energy value and resolution at that particular value of $I = 1.0$ A. The energy spectrum of this representative set of electrons is shown in Figure A.6a and indicates a central energy value of $E = 0.76$ keV with an uncertainty, ΔE, equal to the full-width at half-maximum (FWHM) of 63 eV. The resolution is the quotient of these two quantities and has a value of $\Delta E/E = 8.3$ %. Also shown are the energy spectra for larger values of a and d equal to 1.5 mm and 2.0 mm. As the diaphragm widths are increased, ΔE increases to 9.8% for $a = d = 1.5$ mm and 11.3% for $a = d = 2.0$ mm. Figure A.6b illustrates that E varies quadratically with I up to 50 keV and agrees well with the analytic scaling relationship between applied magnetic field and electron energy for a fixed gyrating radius r_0. Shown in Figure A.6c is the relationship between ΔE and I, which also quadratic in functional form. The resolution, $\Delta E/E$, is obtained through the quotient of the two data sets of Figure A.6b and A.6c, and is illustrated in Figure A.6d. It is observed that the device incurs a $\Delta E/E < 10\%$ at nearly all values of E. It should be noted that the maximum I available for generating $B_z(x,y)$ was 8.0 A, corresponding to an electron kinetic energy of ~50 keV and $\Delta E/E = 9\%$. This maximal value was limited only by the power supply employed, however, operating beyond this value of I would require additional measures for heat sinking as prolonged operation resulted in a noticeable rise in temperature ($\Delta T \cong 25$°C).

A.3.2 Electron Multiplier Response

Charge particle devices can be affected by the surrounding magnetic fields. In the present case, the electron multiplier constitutes such a device and its performance in a magnetic

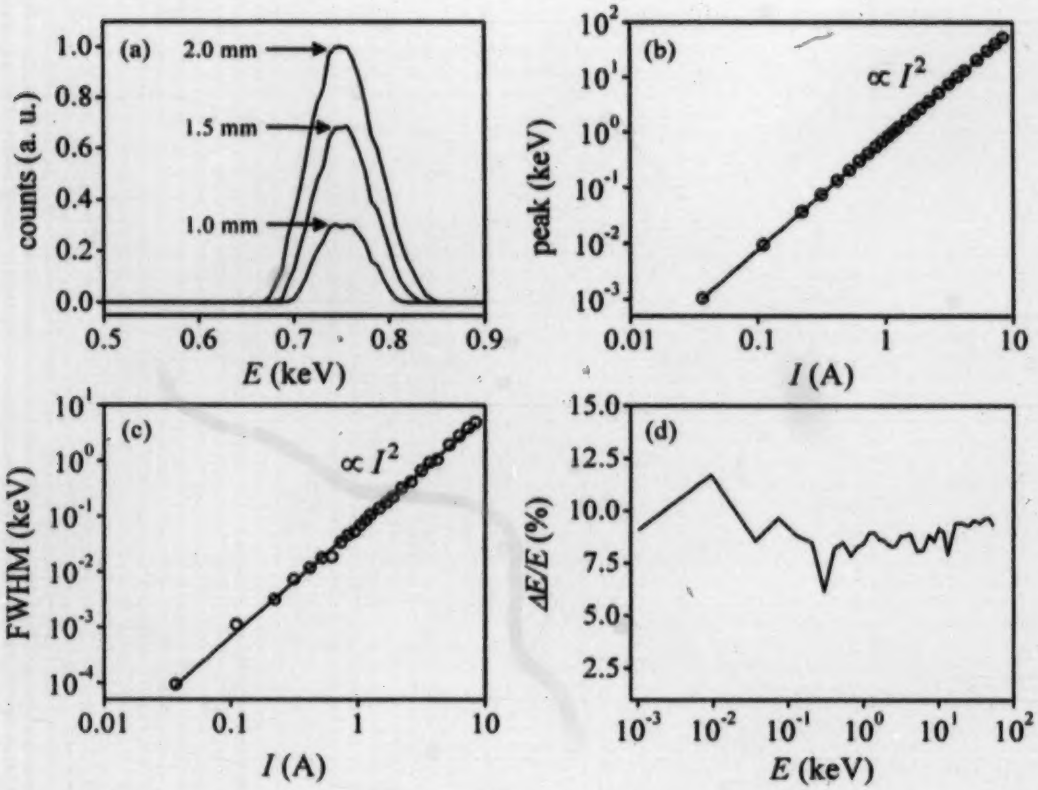

Figure A.6. (a) Transmitted electron energy distribution for $I = 1.0$ A and $s = 2$ mm. Three impulse responses are shown for $a = d = 1.0$ mm, $a = d = 1.5$ mm, and $a = d = 2.0$ mm. For the 1.0 mm case, the central energy $E = 0.76$ keV and FWHM $\Delta E = 63$ eV, resulting in a resolution $\Delta E/E = 8.3$ %. For the increased values of $a = d = 1.5$ mm, and $a = d = 2.0$ mm, the uncertainty in energy measurement increases to 9.8% and 11.3%, respectively. (b) Calculated variation of the central value of energy, E, as a function of the current I (circles) for $a = d = 1.0$ mm and $s = 2$ mm. The least-squares fit (line) indicates a quadratic relationship. (c) Calculated variation of the FWHM, ΔE, as a function of the current I (circles) for $a = d = 1.0$ mm and $s = 2$ mm. The least-squares fit (line) also indicates a quadratic relationship. (d) Overall resolution of the spectrometer for $a = d = 1.0$ mm and $s = 2$ mm.

field must be evaluated since it is placed directly between the generation coils. Previous studies indicate that single or multi-channel, discrete or continuous dynode-type multipliers are all affected by magnetic fields and their orientation with respect to the electron multiplier [14-17]. This does not, however, precluded their usage in the magnetic spectrometer presented here. A titanium-sapphire laser amplifier capable of delivering 30 fs, 800 nm pulses at an intensity of 10 GW/cm^2 is employed to determine the electron multiplier gain as a function of applied magnetic field. Laser light from the amplifier system illuminates the entrance of the multiplier and is used to generate electrons inside the device. Thus, any change in gain results only from a change in $B_z(x,y)$ and is not due to a change in the trajectories of external electrons. Furthermore, special care must be taken to ensure that the orientation of the electron multiplier does not change between calibration and employment in the measurement of an electron energy spectrum. Figure A.7 illustrates the relationship between $B_z(x,y)$ and the gain of the multiplier for magnetic fields ranging up to 35 mT. Here, it is observed that the gain is reduced by over two orders of magnitude as the magnetic field is varied from 0 to 35 mT and is in general agreement with data presented for a similar continuous dynode electron multiplier [17]. The true electron energy spectrum can be deconvolved from the response of the detector using this calibration curve.

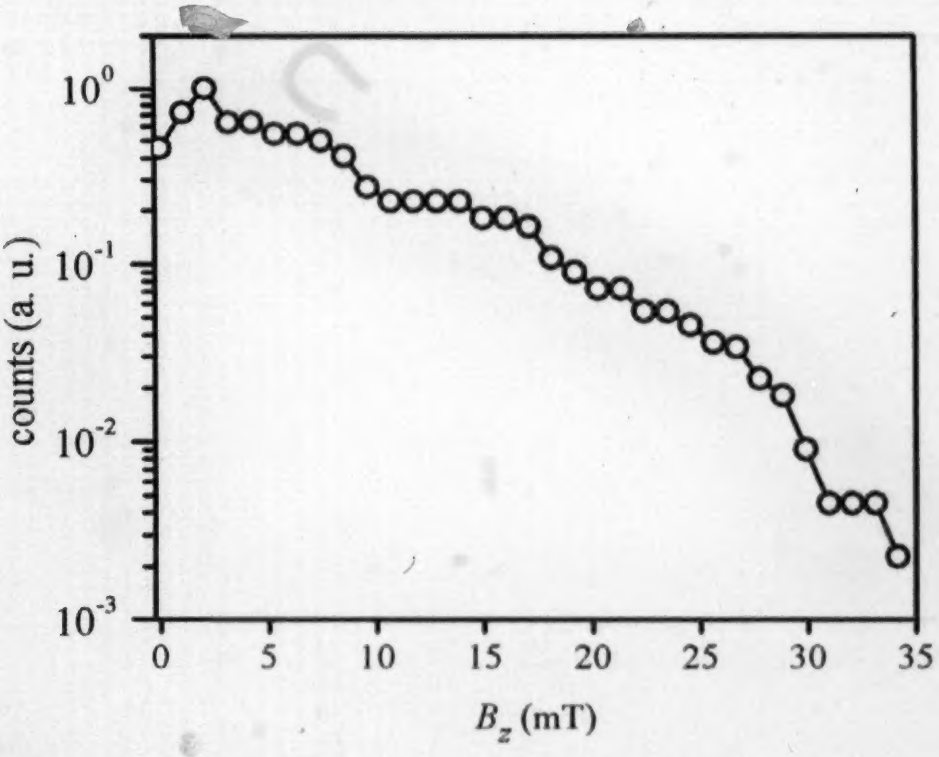

Figure A.7. Electron multiplier gain as a function of the externally applied magnetic field.

A.4 References

[1] D. P. Woodruff and T. A. Delchar, *Modern techniques of surface science* (Cambridge University Press, New York, 1986)

[2] W. Steckelmacher, "Energy analyzers for charged-particle beams," *Journal of Physics E*, vol. **6**, pp. 1061-1071 (1973).

[3] O. Klemperer, "Electron beam spectroscopy," *Reports on Progress in Physics*, vol. **28**, pp. 77-111 (1965).

[4] R. E. Kennerly, "High-resolution pulsed electron-beam time-of-flight spectrometer," *Review of Scientific Instruments*, vol. **48**, pp. 1682-1688 (1977).

[5] R. Z. Bachrach, F. C. Brown, S. B. M. Hagstrom, "Photoelectron-spectroscopy by time-of-flight technique using synchrotron radiation," *Journal of Vacuum Science and Technology*, vol. **12**, pp. 309-312 (1975).

[6] R. F. Schneider, C. M. Luo, M. J. Rhee, and J. R. Smith, "Compact magnetic electron energy analyzer," *Review of Scientific Instruments*, vol. **56**, pp. 1534-1536 (1985).

[7] Y. Okano, Y. Hironaka, K. G. Nakamura, K. Kondo, Y. Oishi, T. Nayuki, and K. Nemoto, "Energy distribution of electrons ejected from a copper target in a femtosecond laser field of 10^{17} W/cm^2," *Journal of Applied Physics*, vol. **95**, pp. 2278-2282 (2004).

[8] K. D. Sevier, *Low Energy Electron Spectrometry* (Wiley-Interscience, New York, 1972).

[9] W. Denk and T. von Egidy, "Design Of An Electrostatic Electron Spectrometer," *Nuclear Instruments and Methods in Physics Research A*, vol. **102**, pp. 281-284 (1972).

[10] H. M. van Noort and A. A. van Gorkum, "Design of a new retarding-field electron spectrometer for depth-selective conversion electron Mössbauer spectroscopy," *Journal of Physics E*, vol. **21**, pp. 587-591 (1988).

[11] D. Varga, I. Kadar, S. Ricz, J. Vegh, A. Kover, B. Sulik, and D. Berényi, "A spherical mirror-double cylindrical mirror electron spectrometer for simultaneous energy and angular distribution measurements: design, construction and experiences," *Nuclear Instruments and Methods in Physics Research A*, vol. **313**, pp. 163-172 (1992).

[12] S. E. Irvine and A. Y. Elezzabi, "Ultracompact 180° magnetic spectrometer for intermediate energy electron measurement," *Measurement Science and Technology*, vol. **17**, pp. 2455-2460 (2006).

[13] S. C. Chappra and R. P. Canale, *Numerical Methods for Engineers* (McGraw-Hill, New York, 1988).

[14] V. V. Anashin, P. M. Beschastnov, V. B. Golubev, L. A. Mironenko, A. A. Salnikov, and S. I. Serednyakov, "Photomultipliers with microchannel plates," *Nuclear Instruments and Methods in Physics Research A*, vol. **357**, pp 103-109 (1995)

[15] P. Bourgeois, M. Karolak, and G. Vasseur, "Performance of the photomultiplier tubes used in the DIRC of BABAR: effect of a magnetic field and of helium," *Nuclear Instruments and Methods in Physics Research A*, vol. **442**, pp. 105-112 (2000).

[16] M. Hashimoto and T. Hayashi, "Effect of Longitudinal Magnetic-Field on the Performance of a Channel Electron Multiplier," *Review of Scientific Instruments*, vol. **44**, pp. 1125-1126 (1973).

[17] T. L. Chenevert and P. W. Zitzewitz, "Performance of a spiralled continuous electron multiplier in a magnetic-field," *Review Scientific Instruments*, vol. **48**, pp. 1601-1602 (1977).

Appendix B

Model Implementation and Source Code

The model described in Chapter 3 is implemented using Visual C++ and the flow chart is depicted in Figure B.1. Essentially, each object is designed to handle either light-wave propagation or electron dynamics and numerous functions are used to communicate between the various program modules. Table B.1 lists each class and provides a brief description of its function. Also shown in Table B.2 are the files associated with each class, which are reproduced individually in the following section.

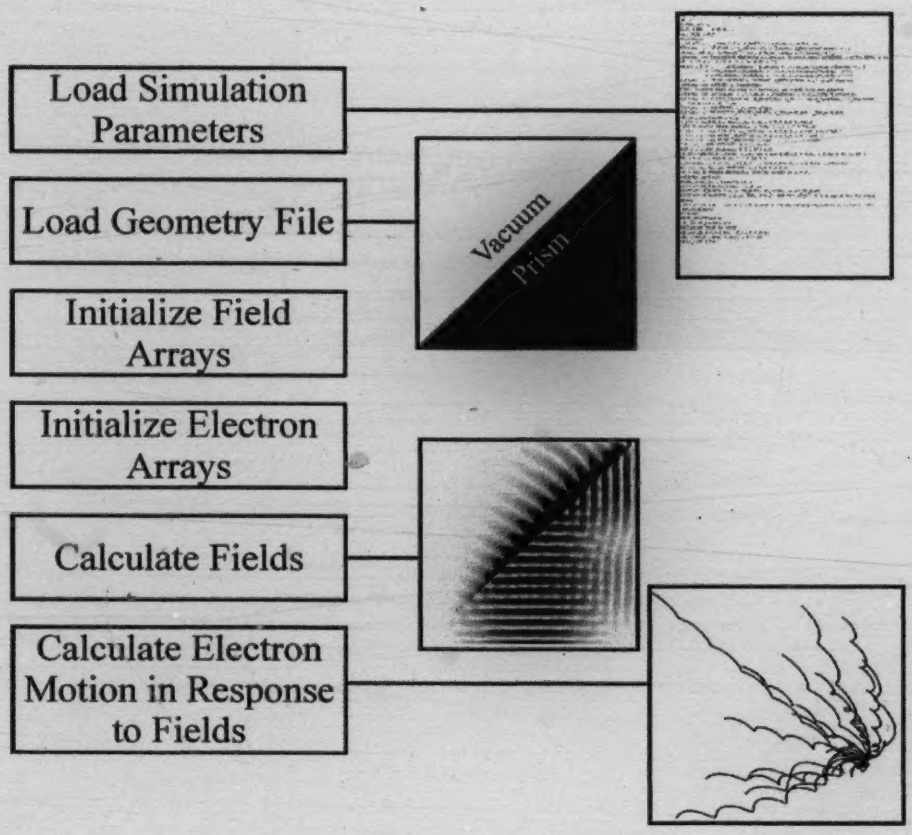

Figure B.1. Flow chart of the simulation program.

Class	Description
Bitmap_RGB	Container class designed to hold and retrieve red, green, and blue values for bitmap handling.
BitmapFieldWriter	Handles output of the electric and magnetic fields to bitmap or text file formats.
BitmapGeometryReader	Inputs the simulation geometry of the FDTD simulation from an external bitmap file.
Electron_Group	Handles a group of Electron_Motion objects.
Electron_Motion	Container class designed to hold information about electron trajectories and predict their motion.
Electron_Source_Finder	Locates the spatial coordinates where electron emission will take place.
Electron_Source_Writer	Outputs a bitmap file that contains the coordinates of an electron source.
FDTDSimulation	Contains the main function and is the entrance of the program.
FDTDSolver	Initializes material parameters and the geometry of the simulation.
GuassianSource	Provides a class for determining a continuous electromagnetic wave source.
Geometry	Contains information of the interaction geometry within the computational window.
GeometryPoint	Container class designed to store information regarding the material properties at a particular grid point.
IntensityDependentLookUp	Calculates the electron emission probability based on an external data set. This class is inherited from Source.
Material	Contains the data for various materials used for the simulation.
Options	Contains the various options used for the simulation, suh as frame spacing, simulation name, etc.
PMLBC	Calculates and stores the perfectly matched layer coefficients.
PropertyReader	Inputs the material properties from an external text file.
PulsedGaussianSource	Provides a class for determining a pulsed wave source. This class is inherited from GaussianSource.
PulsedSecondOrderSolver	This module performs the actual time-stepping of the fields and is inherited from SecondOrderSolver.
SecondOrderSolver	This class remains a virtual one and is used only with PulsedSecondOrderSolver. Extra code can be added if a pulsed solver is not required.
Source	Generic source class that contains data members common to all sources.

Table B.1. Various classes used in the simulation and a brief description of their function.

Class	Associated Files
Bitmap_RGB	Bitmap_RGB.h, Bitmap_RGB.cpp
BitmapFieldWriter	BitmapFieldWriter.h, BitmapFieldWriter.cpp
BitmapGeometryReader	BitmapGeometryReader.h, BitmapGeometryReader.cpp
Electron_Group	Electron_Group.h, Electron_Group.cpp
Electron_Motion	Electron_Motion.h, Electron_Motion.cpp
Electron_Source_Finder	Electron_Source_Finder.h, Electron_Source_Finder.cpp
Electron_Source_Writer	Electron_Source_Writer.h, Electron_Source_Writer.cpp
FDTDSimulation	FDTDSimulation.cpp
FDTDSolver	FDTDSolver.h, FDTDSolver.cpp
GaussianSource	GaussianSource.h, GaussianSource.cpp
Geometry	Geometry.h, Geometry.cpp
GeometryPoint	GeometryPoint.h, GeometryPoint.cpp
IntensityDependentLookUp	IntensityDependentLookUp.h, IntensityDependentLookUp.cpp
Material	Material.h, Material.cpp
Options	Options.h, Options.cpp
PMLBC	PMLBC.h, PMLBC.cpp
PropertyReader	PropertyReader.h, PropertyReader.cpp
PulsedGaussianSource	PulsedGaussianSource.h, PulsedGaussianSource.cpp
PulsedSecondOrderSolver	PulsedSecondOrderSolver.h, PulsedSecondOrderSolver.cpp
SecondOrderSolver	SecondOrderSolver.h, SecondOrderSolver.cpp
Source	Source.h, Source.cpp

Table B.2. Various classes used in the simulation and associated files.

B.1 Program Code

Bitmap_RGB.h

```
#ifndef BITMAP_RGB_H
#define BITMAP_RGB_H

class Bitmap_RGB
{
public:
        Bitmap_RGB();
        Bitmap_RGB(int R, int G, int B);
        void set(int R, int G, int B);
        int get_R();
        int get_G();
        int get_B();
        void operator=(Bitmap_RGB x);
        virtual ~Bitmap_RGB();
private:
        int m_red;
        int m_green;
        int m_blue;

};

#endif
```

Bitmap_RGB.cpp

```
#include "Bitmap_RGB.h"

#ifdef _DEBUG
#undef THIS_FILE
static char THIS_FILE[]=__FILE__;
#define new DEBUG_NEW
#endif

Bitmap_RGB::Bitmap_RGB()
{
		m_red=0;
		m_green=0;
		m_blue=0;

}
Bitmap_RGB::Bitmap_RGB(int R, int G, int B)
{
		m_red=R;
		m_green=G;
		m_blue=B;

}
void Bitmap_RGB::set(int R, int G, int B)
{
		m_red=R;
		m_green=G;
		m_blue=B;

}
int Bitmap_RGB::get_R(){return m_red;}
int Bitmap_RGB::get_G(){return m_green;}
int Bitmap_RGB::get_B(){return m_blue;}
void Bitmap_RGB::operator=(Bitmap_RGB x){
		m_red=x.get_R();
		m_green=x.get_G();
		m_blue=x.get_B();
}
Bitmap_RGB::~Bitmap_RGB(){}
```

BitmapFieldWriter.h

```
#include "Options.h"
#include "Electron_Group.h"

#ifndef BITMAP_FIELD_WRITER_H
#define BITMAP_FIELD_WRITER_H

class BitmapFieldWriter
{

public:
		static int write( char* filename, double** field, int lengthI, int lengthJ, Options* options, int ** electron_locations, double ** electron_numbers);
};

#endif
```

BitmapFieldWriter.cpp

```
#include "BitmapFieldWriter.h"
#include "stdheader.h"
#include "bitmap.h"
#include "options.h"
#include <stdio.h>
#include <math.h>
#include <iostream.h>
#include "Electron_Group.h"

int BitmapFieldWriter::write( char* filename, double** field, int lengthI, int lengthJ, Options* options,int ** electron_locations,double ** electron_numbers)
{
		int rc = SUCCESS;
		int i,j;
		char textfilename[512];
		char bmpfilename[512];
		int mode = options->getOutputMode();

		sprintf(textfilename,"%s.txt",filename);
		sprintf(bmpfilename,"%s.bmp",filename);

		BitmapInfoHeader bih;
		BitmapFileHeader bfh;

		bfh.bfOffBits = sizeof(bfh) + sizeof(bih);
		bfh.bfReserved1 = 0;
		bfh.bfReserved2 = 0;
		bfh.bfType = 19778;
		bfh.bfSize = bfh.bfOffBits + (lengthI*lengthJ)*3;
```

```
bih.biSize = sizeof(bih);
bih.biBitCount=24;
bih.biHeight=lengthJ;
bih.biWidth=lengthI;
bih.biCompression=0;
bih.biPlanes=1;
bih.biXPelsPerMeter=3780;
bih.biYPelsPerMeter=3780;
bih.biSizeImage = (lengthJ*lengthI)*3;
bih.biClrUsed = 0;
bih.biClrImportant =0;
FILE *bfp = NULL;
FILE *fp = NULL;

if(mode==BMP_OUTPUT_MODE||mode==BMP_TXT_OUTPUT_MODE){
        bfp=fopen(bmpfilename,"wb");
        if(bfp==NULL){
                rc = INVALID_FILENAME;
                return rc;
        }
}


if(mode==TXT_OUTPUT_MODE||mode==BMP_TXT_OUTPUT_MODE){
        fp=fopen(textfilename,"wt");
        if(fp==NULL){
                rc = INVALID_FILENAME;
                return rc;
        }
}

if(mode==BMP_OUTPUT_MODE||mode==BMP_TXT_OUTPUT_MODE){
        fwrite(&bfh,sizeof(bfh),1,bfp);
        fwrite(&bih,sizeof(bih),1,bfp);
}

double min=options->getMinOutputField();
double max=options->getMaxOutputField();

for(i=0;i<lengthJ;i++){
        for( j=0;j<lengthI;j++){
                if(mode==BMP_OUTPUT_MODE||mode==BMP_TXT_OUTPUT_MODE){
                        unsigned char bgr[3];
                        int colorRange = (5*256);
                        double percent = (fabs(min)+field[j][i])/(fabs(max)+fabs(min) );
                        int color = percent*colorRange;

                        bgr[0] = color <= 128 ? 255
                                : ( color <= 3*256/2 ? 255
                                : ( color <= 5*256/2 ? 5*256/2 - color
                                : ( color < 7*256/2 ? 0
                                : ( color < 9*256/2 ? 0
                                : color-9*256/2 ) ) ) );
                        bgr[1] = color <= 128 ? 255
                                : ( color <= 3*256/2 ? 3*256/2 - color
                                : ( color <= 5*256/2 ? 0
                                : ( color < 7*256/2 ? 0
                                : ( color < 9*256/2 ? color - 7*256/2
                                : 255 ) ) ) );
                        bgr[2] = color <= 128 ? 128-color
                                : ( color <= 3*256/2 ? 0
                                : ( color <= 5*256/2 ? 0
                                : ( color < 7*256/2 ? color-5*256/2
                                : ( color < 9*256/2 ? 255
                                : 255 ) ) ) );

                        if(electron_locations[j][i]==1){

                                double percent =( 1 + 2*electron_numbers[j][i] )/( 1 + 1 );
                                int color = percent*colorRange;
                                bgr[0] = color <= 128 ? 255
                                        : ( color <= 3*256/2 ? 255
                                        : ( color <= 5*256/2 ? 5*256/2 - color
                                        : ( color < 7*256/2 ? 0
                                        : ( color < 9*256/2 ? 0
                                        : color-9*256/2 ) ) ) );
                                bgr[1] = color <= 128 ? 255
                                        : ( color <= 3*256/2 ? 3*256/2 - color
                                        : ( color <= 5*256/2 ? 0
                                        : ( color < 7*256/2 ? 0
                                        : ( color < 9*256/2 ? color - 7*256/2
                                        : 255 ) ) ) );
                                bgr[2] = color <= 128 ? 128-color
                                        : ( color <= 3*256/2 ? 0
                                        : ( color <= 5*256/2 ? 0
                                        : ( color < 7*256/2 ? color-5*256/2
                                        : ( color < 9*256/2 ? 255
                                        : 255 ) ) ) );
                        }
                        fwrite(bgr,sizeof(unsigned char),3,bfp);
                }
```

```
                                        if(mode==TXT_OUTPUT_MODE
                                                ||mode==BMP_TXT_OUTPUT_MODE){
                                                fprintf(fp," %e",field[j][i]);
                                        }
                                }

                if(j%4!=0&&(mode==BMP_OUTPUT_MODE
                                ||mode==BMP_TXT_OUTPUT_MODE))
                {
                                        unsigned char space = 0;
                                        int n=0;
                                        for(n=0;n<j%4;n++){
                                                fwrite(&space,sizeof(unsigned char),1,bfp);
                                        }
                                }

                                if(mode==TXT_OUTPUT_MODE||mode==BMP_TXT_OUTPUT_MODE){
                                        fprintf(fp,"\n");
                                }
                }

                if( fp != NULL ){fclose(fp);}
                if( bfp != NULL ){fclose(bfp);}
                return rc;
}
```

BitmapGeometryReader.h

```
#ifndef BITMAP_GEOMETRY_READER_H
#define BITMAP_GEOMETRY_READER_H

#include "Geometry.h"

class BitmapGeometryReader
{
private:
                Geometry *geometry;

public:
                virtual int readGeometry( char* geomFile, Geometry* geometry);
};

#endif
```

BitmapGeometryReader.cpp

```
#include "BitmapGeometryReader.h"
#include "PMLBC.h"
#include "GaussianSource.h"
#include "Bitmap.h"
#include <cstdio>
#include <iostream.h>
#include "Geometry.h"
#include <assert.h>

/** Reads in the geometry from a 24 bit bitmap file
 *  Returns 0 on success, >0 on failure.
 */
int BitmapGeometryReader::readGeometry( char *geomFile, Geometry* geom){
                FILE* geomFP;
                BitmapFileHeader bfh;
                BitmapInfoHeader bih;
                int bytesRead = 0;
                BYTE_1* bytesReversed = NULL;
                long imageSize;
                int i,j,n;
                int rc = SUCCESS;
                geometry = geom;
                PMLBC *bc =PMLBC::getBoundaryCondition();

                //make sure we have a boundary condition!!
                assert( bc != NULL );

                //Open the geometry files for reading
                geomFP = fopen( geomFile, "rb" );
                if( geomFP == NULL ){
                                cerr << "Geometry file not found!\n";
                                return FILE_NOT_FOUND;
                }

                //Read in the geometry from the bitmap file
                bytesRead = fread(&bfh,1,sizeof(BitmapFileHeader),geomFP );
                if( bytesRead != sizeof(BitmapFileHeader) ){
                                //cerr << "Bytes Read: " << bytesRead << "\n";
                                //cerr << "Size of header: " << sizeof(BitmapFileHeader) << "\n";
                                cerr << "Incorrect geometry file header format!\n";
                                fclose( geomFP );
                                return INVALID_FILE_FORMAT;
                }
```

```
        bytesRead = fread(&bih,1,sizeof(BitmapInfoHeader),geomFP );
        if( bytesRead != sizeof(BitmapInfoHeader ) ){
                cerr << "Incorrect geometry info header format!\n";
                fclose( geomFP );
                return INVALID_FILE_FORMAT;
        }

        //must be using a 24 bit bitmap
        if( bih.biBitCount != BIT_DEPTH ){
                cerr << "Incorrect geometry format. Bit detph != 24...\n";
                fclose( geomFP );
                return INVALID_FILE_FORMAT;
        }

        //read in the bytes from the file
        geometry->m_iWidth = bih.biWidth;
        geometry->m_iHeight = bih.biHeight;
        imageSize = BIT_DEPTH*geometry->m_iHeight*geometry->m_iWidth
                / BITS_PER_BYTE + (geometry->m_iWidth*3) % 4*geometry
                ->m_iHeight;

        bytesReversed = new BYTE_1[imageSize];
        bytesRead = fread(bytesReversed,sizeof(BYTE_1),imageSize,geomFP);
        if( bytesRead != imageSize ) {
                cerr << "Error reading file information!\n";
                fclose( geomFP );
                return INVALID_FILE_FORMAT;
        }

        //populate the geometry points
        geometry->points = new GeometryPoint*[geometry->m_iHeight];
        n=0;
        for(j=0;j<geometry->m_iHeight&&rc==SUCCESS;j++){
                geometry->points[j] = new GeometryPoint[geometry->m_iWidth];
                for(i=0;i<geometry->m_iWidth&&rc==SUCCESS;i++){
                        long index = Material::getIndexForColor(bytesReversed[n],bytesReversed[n+1],bytesReversed[n+2]);
                        n+=3;

                        Material* mat = Material::materialExists( index );
                        if( mat != NULL ){
                                geometry->points[j][i].setMaterial( mat );
                                if( mat->isSource() ){
                                        Source* source = (Source*) mat;
                                        source->setPoint( i, j );
                                        geometry->setHorizontalSourceLocation(j);
                                        mat->setElectricPermittivity(
                                                geometry->getPoints()[j-1][i].getMaterial()
                                                ->getElectricPermittivity());
                                }
                        }
                        else{
                                cerr << "Error: Material for color bgr " <<(unsigned int)bytesReversed[n] << " "
                                <<(unsigned int)bytesReversed[n+1] << " "<< (unsigned int)bytesReversed[n+2]
                                <<" was null at position x=" << i << " y=" << j << "\n";
                                rc = INVALID_FILE_FORMAT;
                                break;
                        }
                }
                n+=(n%4);
        }

        delete [] bytesReversed;
        fclose( geomFP );
        return rc;
}
```

Electron_Group.h

```
#include "Geometry.h"
#include "Electron_Motion.h"
#include <stdlib.h>
#include <fstream.h>

#ifndef ELECTRON_GROUP_H
#define ELECTRON_GROUP_H

class Electron_Group
{
private:
        int number_of_electrons;
        int group_number;
        int x_location;
        int y_location;
        Electron_Motion *electrons;
        ofstream *output_files;
        double center_time;
        bool create_files;

public:
        Electron_Group();
```

```
		int getNumber();
		Electron_Motion getElectron(int index);
		Electron_Motion *getElectronArrayPointer(){return electrons;}
		int getX(){return x_location;}
		int getY(){return y_location;}
		double getCenterTime(){return center_time;}
		void setCreateFiles(bool value){create_files=value;}
		ofstream *getFileArrayPointer(){return output_files;}
		void Forward_Iterate(double ** electric_field_x, double ** electric_field_y,double ** magnetic_field_z,double ** previous_electric_field_x,
			double ** previous_electric_field_y,double ** previous_magnetic_field_z,Geometry *geometry,int current_time_step);
		void SetInitialConditions(int number,int group_index,int x_pos,int y_pos,Geometry *geometry);
		virtual ~Electron_Group();
};

#endif
```

Electron_Group.cpp

```
#include "Electron_Group.h"
#include "PulsedGaussianSource.h"
#include "Material.h"
#include "Math.h"
#include "stdheader.h"
#include <stdio.h>
#include <iostream>
#include "IntensityDependentLookUp.h"

Electron_Group::Electron_Group()
{
		number_of_electrons=0;
		group_number=0;
		x_location=0;
		y_location=0;
		electrons=NULL;
		output_files=NULL;
		create_files=false;
}

int Electron_Group::getNumber(){return number_of_electrons;}

Electron_Motion Electron_Group::getElectron(int index){return electrons[index];}

void Electron_Group::Forward_Iterate(double ** electric_field_x,double ** electric_field_y,double ** magnetic_field_z,double ** previous_electric_field_x,
			double ** previous_electric_field_y,double ** previous_magnetic_field_z,Geometry *geometry,int current_time_step){
		for(int i=0;i<number_of_electrons;i++){
			electrons[i].Forward_Iterate(electric_field_x, electric_field_y,magnetic_field_z, previous_electric_field_x, previous_electric_field_y,
			previous_magnetic_field_z, geometry,current_time_step);
		}
		for(i=0;i<number_of_electrons;i++){
			output_files[i]
			<<(current_time_step+1)*geometry->getDeltaTime()<<" "
			<<electrons[i].getX()<<" "<<electrons[i].getY()<<" "<<electrons[i].getVx()
			<<" "<<electrons[i].getVy()<<" "<<electrons[i].getNumber()<<endl;
		}
}


Electron_Group::~Electron_Group(){
		if(electrons!=NULL){
			delete [] electrons;
		}
		if(output_files!=NULL){
			for(int i=0;i<number_of_electrons;i++){
				output_files[i].close();
			}
			delete [] output_files;
		}

}
void Electron_Group::SetInitialConditions(int number, int group_index, int x_pos, int y_pos,Geometry *geometry){

		Material *mat1=Material::getMaterials();
		double pulseDuration;
		double pulseFWHM;
		double sourceRefractiveIndex;
		double start_time;
		x_location=x_pos;
		y_location=y_pos;
		int i;

		number_of_electrons=number;
		group_number=group_index;
		electrons = new Electron_Motion[number_of_electrons];
		output_files = new ofstream[number_of_electrons];
		char filename[512];

		while (mat1 != NULL)
		{
			PulsedGaussianSource *pgs = dynamic_cast<PulsedGaussianSource*>(mat1);
			if( pgs != NULL){
				pulseDuration = pgs->getPulseDuration();
```

```
                                pulseFWHM = pgs->getFWHM();
                                sourceRefractiveIndex = pgs->getRefractiveIndex();
                                break;
                        }
                        mat1 = mat1->nextMaterial;
                }

                //open output files
                if(create_files){
                        for(i=0;i<number_of_electrons;i++){
                                sprintf(filename,"electron_trajectory_group%d_electron_%d.txt", group_index,i);
                                output_files[i].open(filename);
                        }
                }

                double pulseSigma = pulseDuration / ( 2 * sqrt( log( 2 ) ) );
                double sigma_FWHM = pulseFWHM / ( 2 * sqrt( log( 2 ) ) );
                double intensity=0,electron_numbers=0;
                center_time=(y_pos-geometry->getHorizontalSourceLocation())*geometry->getDeltaSpace()*sourceRefractiveIndex/LIGHT_SPEED+2*pulseDuration;

//              need this for the intensity dependent photoemission
/*              IntensityDependentLookUp current_value;
                current_value.Set_Array_Length(901);
                current_value.Initialize_Arrays();
*/

                for(i=0;i<number_of_electrons;i++){
                        start_time=center_time-pulseDuration+2*pulseDuration*((double)i)/((double) number_of_electrons);
                        intensity=exp(-pow((center_time-start_time)/pulseSigma,2))*exp(-pow((x_pos-geometry->getGridWidth()/2)
                                *geometry->getDeltaSpace()/sigma_FWHM,2));
                        electron_numbers=pow(intensity,3);
//                      electron_numbers=current_value.Get_Current(intensity);

                        electrons[i].SetInitialConditions(x_pos*geometry->getDeltaSpace(), y_pos*geometry->getDeltaSpace(),0.0,0.0,start_time, electron_numbers,false);
                }
                if(create_files){
                        for(i=0;i<number_of_electrons;i++){
                                output_files[i] <<0.0<<" "<<electrons[i].getX()<<" "<<electrons[i].getY()<<" "
                                <<electrons[i].getVx()<<" "<<electrons[i].getVy()<<" "<<electrons[i].getNumber()<<endl;
                        }
                }
}
```

Electron_Motion.h

```
#include "Geometry.h"
#ifndef ELECTRON_MOTION_H
#define ELECTRON_MOTION_H

class Electron_Motion
{
private:
        double number_of_electrons;
        double x_position;
        double y_position;
        double x_velocity;
        double y_velocity;
        double start_time;
        bool recaptured;
        bool active;
        double max_electric_field;

public:
        Electron_Motion();
        Electron_Motion(double initial_x,double initial_y,double initial_vx,double initial_vy,double st,double number,bool isActive);
        void Forward_Iterate(double ** electric_field_x, double ** electric_field_y, double ** magnetic_field_z, double ** previous_electric_field_x,
                double ** previous_electric_field_y,double ** previous_magnetic_field_z,Geometry *geometry,int current_time_step);
        double getX();
        double getY();
        double getVx();
        double getVy();
        double getStartTime();
        double getNumber();
        bool isRecaptured();
        bool isActive();
        void SetInitialConditions(double initial_x,double initial_y,double initial_vx,double initial_vy,double st,double number,bool isActive);
        void SetNumber(double number){number_of_electrons=number;}
        virtual ~Electron_Motion();
};

#endif //
```

Electron_Motion.cpp

```
#include "Electron_Motion.h"
#include "stdheader.h"
#include "Geometry.h"
#include <iostream.h>
#include "Material.h"
```

```
#include "PulsedGaussianSource.h"

Electron_Motion::Electron_Motion()
{
	x_position=0.0;
	y_position=0.0;
	x_velocity=0.0;
	y_velocity=0.0;
	start_time=0.0;
	number_of_electrons=0;
	recaptured=false;
	active=false;
	Material *mat1=Material::getMaterials();

	while (mat1 != NULL)
	{
		PulsedGaussianSource *pgs = dynamic_cast<PulsedGaussianSource*>(mat1);
		if( pgs != NULL)
		{
			max_electric_field=pgs->getMaxElectricField();
			break;
		}
		mat1 = mat1->nextMaterial;
	}

}
Electron_Motion::Electron_Motion(double initial_x,double initial_y,double initial_vx,double initial_vy,double st,double number,bool isActive)
{
	x_position=initial_x;
	y_position=initial_y;
	x_velocity=initial_vx;
	y_velocity=initial_vy;
	start_time=st;
	number_of_electrons=number;
	recaptured=false;
	active=isActive;

}
void Electron_Motion::Forward_Iterate(double ** electric_field_x, double ** electric_field_y, double ** magnetic_field_z,
		double ** previous_electric_field_x, double ** previous_electric_field_y, double ** previous_magnetic_field_z,
		Geometry *geometry, int current_time_step){
	if((current_time_step*geometry->getDeltaTime())<start_time){
		return;
	}
	active=true;

	double xprev=x_position;
	double yprev=y_position;
	double vxprev=x_velocity;
	double vyprev=y_velocity;

	x_position=geometry->getDeltaTime()*x_velocity+x_position;
	y_position=geometry->getDeltaTime()*y_velocity+y_position;

	int current_i,current_j,rounded_i,rounded_j;
	int shift_i=0,shift_j=0;

	double dx=geometry->getDeltaSpace();
	double shift_x=-0.5*dx,shift_y=-0.5*dx;

	current_i=(int)(x_position/geometry->getDeltaSpace());
	current_j=(int)(y_position/geometry->getDeltaSpace());
	rounded_i=(int)(x_position/geometry->getDeltaSpace()+0.5);//the 0.5 is for rounding
	rounded_j=(int)(y_position/geometry->getDeltaSpace()+0.5);//the 0.5 is for rounding

	if(current_i==rounded_i){
		shift_j=1;
		shift_y*=-1.0;
	}
	if(current_j==rounded_j){
		shift_i=1;
		shift_x*=-1.0;
	}

	Material *mat;
	if( (current_i<0)||(current_i>=geometry->getGridWidth())||(current_j<0)||(current_j>=geometry->getGridHeight()) ){
		int location_i=0;
		int location_j=0;
		if(current_i<0){location_i=0;}
		if(current_i>=geometry->getGridWidth()){location_i=geometry->getGridWidth()-1;}
		if(current_j<0){location_j=0;}
		if(current_j>=geometry->getGridHeight()){location_j=geometry->getGridHeight()-1;}
		mat = geometry->getMaterial(location_i,location_j);
	}
	else{
		mat = geometry->getMaterial(current_i,current_j);
	}


	int time_check=(int)(start_time/geometry->getDeltaTime()+0.5);

	if(mat->isMetal()){recaptured=true;}
	if(mat->isConductive()){recaptured=true;}
```

```
/*convention*********************************/
//The hz field is at the actual points in space and time where as the electric
//field is on the half points.
/***********************************************************/

        //check boundaries

        double y1,y2,y3,y4,tt,u;
        double ex_prev,ex,ey_prev,ey,mz;
        //electric field x, n-0.5
        if((current_i<0) ||(current_i>=geometry->getGridWidth()) ||(current_i+1<0) ||(current_i+1>=geometry->getGridWidth())
                ||(current_j-shift_i<0) || (current_j-shift_i>=geometry->getGridHeight()) || (current_j+1-shift_i<0) ||
                (current_j+1-shift_i>=geometry->getGridHeight())){
                ex_prev=0;
        }
        else {
                y1=previous_electric_field_x[current_i][current_j-shift_i];
                y2=previous_electric_field_x[current_i+1][current_j-shift_i];
                y3=previous_electric_field_x[current_i+1][current_j+1-shift_i];
                y4=previous_electric_field_x[current_i][current_j+1-shift_i];
                tt=(x_position-current_i*dx)/dx;
                u=(y_position+shift_x-current_j*dx)/dx;
                ex_prev=(1-tt)*(1-u)*y1+tt*(1-u)*y2+tt*u*y3+(1-tt)*u*y4;
        }
        //electric field x, n+0.5
        if((current_i<0) ||(current_i>=geometry->getGridWidth()) ||(current_i+1<0) ||(current_i+1>=geometry->getGridWidth())
                ||(current_j-shift_i<0) || (current_j-shift_i>=geometry->getGridHeight()) || (current_j+1-shift_i<0)
                ||(current_j+1-shift_i>=geometry->getGridHeight())){
                ex=0;
        }
        else{
                y1=electric_field_x[current_i][current_j-shift_i];
                y2=electric_field_x[current_i+1][current_j-shift_i];
                y3=electric_field_x[current_i+1][current_j+1-shift_i];
                y4=electric_field_x[current_i][current_j+1-shift_i];
                tt=(x_position-current_i*dx)/dx;
                u=(y_position+shift_x-current_j*dx)/dx;
                ex=(1-tt)*(1-u)*y1+tt*(1-u)*y2+tt*u*y3+(1-tt)*u*y4;
        }


        //electric field y n-0.5
        if((current_i-shift_j<0) ||(current_i-shift_j>=geometry->getGridWidth()) ||(current_i+1-shift_j<0)
                ||(current_i+1-shift_j>=geometry->getGridWidth()) ||(current_j<0) ||(current_j>=geometry->getGridHeight())
                ||(current_j+1<0) ||(current_j+1>=geometry->getGridHeight())){
                ey_prev=0;
        }
        else{
                y1=previous_electric_field_y[current_i-shift_j][current_j];
                y2=previous_electric_field_y[current_i+1-shift_j][current_j];
                y3=previous_electric_field_y[current_i+1-shift_j][current_j+1];
                y4=previous_electric_field_y[current_i-shift_j][current_j+1];
                tt=(x_position+shift_y-current_i*dx)/dx;
                u=(y_position-current_j*dx)/dx;
                ey_prev=(1-tt)*(1-u)*y1+tt*(1-u)*y2+tt*u*y3+(1-tt)*u*y4;
        }


        //electric field y n+0.5
        if((current_i-shift_j<0) ||(current_i-shift_j>=geometry->getGridWidth()) ||(current_i+1-shift_j<0) ||(current_i+1-shift_j>=geometry->getGridWidth())
                ||(current_j<0) ||(current_j>=geometry->getGridHeight()) ||(current_j+1<0) ||(current_j+1>=geometry->getGridHeight())){
                ey=0;
        }
        else{
                y1=electric_field_y[current_i-shift_j][current_j];
                y2=electric_field_y[current_i+1-shift_j][current_j];
                y3=electric_field_y[current_i+1-shift_j][current_j+1];
                y4=electric_field_y[current_i-shift_j][current_j+1];
                tt=(x_position+shift_y-current_i*dx)/dx;
                u=(y_position-current_j*dx)/dx;
                ey=(1-tt)*(1-u)*y1+tt*(1-u)*y2+tt*u*y3+(1-tt)*u*y4;
        }

        //magnetic field z
        if((current_i<0) || (current_i>=geometry->getGridWidth()) || (current_i+1<0) || (current_i+1>=geometry->getGridWidth()) ||
                (current_j<0) || (current_j>=geometry->getGridHeight()) || (current_j+1<0) || (current_j+1>=geometry->getGridHeight())){
                mz=0;
        }
        else{
                y1=magnetic_field_z[current_i][current_j];
                y2=magnetic_field_z[current_i+1][current_j];
                y3=magnetic_field_z[current_i+1][current_j+1];
                y4=magnetic_field_z[current_i][current_j+1];
                tt=(x_position-current_i*dx)/dx;
                u=(y_position-current_j*dx)/dx;
                mz=(1-tt)*(1-u)*y1+tt*(1-u)*y2+tt*u*y3+(1-tt)*u*y4;
        }
        x_velocity=vxprev+geometry->getDeltaTime()*CHARGE_MASS_RATIO*
                (0.5*ex_prev+0.5*ex
                +MU_NOT*vyprev*mz);
```

```
                y_velocity=vyprev+geometry->getDeltaTime()*CHARGE_MASS_RATIO*
                        (0.5*ey_prev+0.5*ey
                        -MU_NOT*vxprev*mz);

}

double Electron_Motion::getX(){return x_position;}
double Electron_Motion::getY(){return y_position;}
double Electron_Motion::getVx(){return x_velocity;}
double Electron_Motion::getVy(){return y_velocity;}
double Electron_Motion::getStartTime(){return start_time;}
double Electron_Motion::getNumber(){return number_of_electrons;}
bool Electron_Motion::isRecaptured(){return recaptured;}
bool Electron_Motion::isActive(){return active;}

void Electron_Motion::SetInitialConditions(double initial_x,double initial_y,double initial_vx,double initial_vy,double st,double number,bool isActive){
        x_position=initial_x;
        y_position=initial_y;
        x_velocity=initial_vx;
        y_velocity=initial_vy;
        number_of_electrons=number;
        start_time=st;
        active=isActive;
}


Electron_Motion::~Electron_Motion()
{

}
```

Electron_Source_Finder.h

```
#include "Geometry.h"
#ifndef ELECTRON_SOURCE_FINDER_H
#define ELECTRON_SOURCE_FINDER_H

class Electron_Source_Finder
{
private:
        int number_of_sources;
        int *x_positions;
        int *y_positions;
public:
        Electron_Source_Finder(int x_extent,int number_of_electron_groups,
                Geometry* geometry);
        int *get_x_array(){return x_positions;}
        int *get_y_array(){return y_positions;}
        virtual ~Electron_Source_Finder();

};

#endif
```

Electron_Source_Finder.cpp

```
#include "Electron_Source_Finder.h"
#include "Geometry.h"
#include "Material.h"
#include "PulsedGaussianSource.h"
#include <stdio.h>
#include <iostream.h>


Electron_Source_Finder::Electron_Source_Finder(int x_extent,int number_of_electron_groups,Geometry* geometry){
        number_of_sources=number_of_electron_groups;
        x_positions=new int[number_of_sources];
        y_positions=new int[number_of_sources];
        int x0=geometry->getGridWidth()/2;
        int y0=0;
        double sigma=0;
        Material *mat=Material::getMaterials(),*mat1,*mat2;

        while (mat!= NULL){
                PulsedGaussianSource *pgs =
                        dynamic_cast<PulsedGaussianSource*>(mat);
                if( pgs != NULL){
                        sigma = pgs->getPulseFWHM()/ ( 2 * sqrt( log( 2 ) ) );
                        break;
                }
                mat = mat->nextMaterial;
        }

//linear distribution
        for(int i=0;i<number_of_electron_groups;i++){
                x_positions[i]=(int)(x0-(double)x_extent/2.0
                        +((double)i/((double) number_of_electron_groups)
                        *((double) x_extent)));
                for(int j=0;j<geometry->getGridHeight()-1;j++){
```

```
                                        mat1=geometry->getMaterial(x_positions[i],j);
                                        mat2=geometry->getMaterial(x_positions[i],j+1);
                                        if(mat1->isMetal()&&!mat2->isMetal()){
                                                y_positions[i]=j+1;//+1+1;
                                        }//=x_positions[i]+y0-x0;
                                }
                        }
}

Electron_Source_Finder::~Electron_Source_Finder(){
                if(x_positions!=NULL){
                        delete [] x_positions;
                }
                if(y_positions!=NULL){
                        delete [] y_positions;
                }
}
```

Electron_Source_Writer.h

```
#include "Bitmap_RGB.H"
#include "Geometry.h"
#ifndef ELECTRON_SOURCE_WRITER_H
#define ELECTRON_SOURCE_WRITER_H

#if _MSC_VER > 1000
#pragma once
#endif // _MSC_VER > 1000

class Electron_Source_Writer
{
public:
                Electron_Source_Writer(int *x_array,int *y_array, int num_sources,
                        Geometry *geometry);
                virtual ~Electron_Source_Writer();
                int Write( char* filename, Bitmap_RGB** field, int lengthI, int lengthJ);
};

#endif
```

Electron_Source_Writer.cpp

```
#include "Electron_Source_Writer.h"
#include "Bitmap_RGB.h"
#include "bitmap.h"
#include "stdio.h"
#include "fstream.h"
#define SUCCESS 0
#define INVALID_FILENAME 5

Electron_Source_Writer::Electron_Source_Writer(int *x_array,int *y_array, int num_sources,Geometry *geometry)
{
                Bitmap_RGB **profile=new Bitmap_RGB* [geometry->getGridWidth()];

                for(int init=0;init<geometry->getGridWidth();init++){
                        profile[init]=new Bitmap_RGB [geometry->getGridHeight()];
                }

                for(int i=0;i<geometry->getGridWidth();i++){
                        for(int j=0;j<geometry->getGridHeight();j++){
                                                profile[i][j].set(255,255,255);
                        }
                }


                for(i=0;i<num_sources;i++){
                        profile[x_array[i]][y_array[i]-1].set(0,0,0);
                }
                int check=Write("electron_sources",profile,geometry->getGridWidth(),geometry
                        ->getGridHeight());

                for( i=0; i<geometry->getGridWidth(); i++ ){
                        delete [] profile[i];
                }
                delete [] profile;
}

int Electron_Source_Writer::Write( char* filename, Bitmap_RGB** field, int lengthI,
                int lengthJ)
{
                int rc = SUCCESS;
                int i,j;

                char bmpfilename[512];
                sprintf(bmpfilename,"%s.bmp",filename);

                BitmapInfoHeader bih;
                BitmapFileHeader bfh;
```

```
	bfh.bfOffBits = sizeof(bfh) + sizeof(bih);
	bfh.bfReserved1 = 0;
	bfh.bfReserved2 = 0;
	bfh.bfType = 19778;
	bfh.bfSize = bfh.bfOffBits + (lengthI*lengthJ)*3;

	bih.biSize = sizeof(bih);
	bih.biBitCount=24;
	bih.biHeight=lengthJ;
	bih.biWidth=lengthI;
	bih.biCompression=0;
	bih.biPlanes=1;
	bih.biXPelsPerMeter=3780;
	bih.biYPelsPerMeter=3780;
	bih.biSizeImage = (lengthJ*lengthI)*3;
	bih.biClrUsed = 0;
	bih.biClrImportant =0;
	FILE *bfp = NULL;

	bfp = fopen( bmpfilename, "wb" );
	if( bfp == NULL )		{
		rc = INVALID_FILENAME;
		return rc;
	}

	fwrite(&bfh,sizeof(bfh),1,bfp);
	fwrite(&bih,sizeof(bih),1,bfp);

	double min = 1;
	double max = 1;

	for( i=0;i<lengthI;i++){
		for( j=0;j<lengthJ;j++ ){
			unsigned char bgr[3];

			bgr[0] = field[j][i].get_B();
			bgr[1] = field[j][i].get_G();
			bgr[2] = field[j][i].get_R();

			fwrite(bgr,sizeof(unsigned char),3,bfp);
		}

		if( j % 4 != 0 ){
			unsigned char space = 0;
			int n = 0;
			for( n=0; n<j%4 ; n ++ )
				fwrite(&space,sizeof(unsigned char),1,bfp);
		}
	}

	if( bfp != NULL )
		fclose(bfp);
	return rc;
}

Electron_Source_Writer::~Electron_Source_Writer()
{

}
```

FDTDSimulation.cpp

```
#include "Geometry.h"
#include "BitmapGeometryReader.h"
#include "PropertyReader.h"
#include "FDTDSolver.h"
#include "SecondOrderSolver.h"
#include "PulsedSecondOrderSolver.h"
#include "GaussianSource.h"
#include <time.h>
#include <math.h>
#include <stdio.h>
#include <iostream.h>

int main( char* argc, char* argv[] )
{
	int rc;

	time_t startTime;
	time_t endTime;
	time_t runTime;
	startTime = time (NULL);
	Geometry *g = new Geometry();
	Options *options = new Options();

	PropertyReader *pr = new PropertyReader();
	rc = pr->readProperties( "properties.prop", g, options );
	if( rc != 0 ){
		cout << "Error reading in properties! Exiting...\n";
		delete pr;
		delete g;
```

```
                Material::deleteMaterials();
                return 1;
        }
        else
                delete pr;

        BitmapGeometryReader *bgr = new BitmapGeometryReader();
        rc = bgr->readGeometry( "geometry.bmp", g );
        if( rc != 0 ){
                cout << "Error reading in geometry! Exiting...\n";
                delete bgr;
                delete g;
                Material::deleteMaterials();
                return 1;
        }
        else
                delete bgr;

        FDTDSolver *sos = NULL;

                sos = new PulsedSecondOrderSolver( g, options );
                if( sos->initialize() != SUCCESS ){
                        delete sos;
                        delete g;
                        delete options;
                        return rc;
                }

        sos->solve();

        endTime = time (NULL);
        runTime = endTime - startTime;
        FILE* fp = fopen("RunningTime.tim","wt");
        if( fp != NULL ){
                long hrs = (long)(runTime/3600);
                long minutes= (long)((runTime - hrs*3600)/60);
                long seconds= (runTime - hrs*3600 - minutes*60);
                cerr << "Run time: " << hrs << " hrs " << minutes << " minutes " <<seconds << " seconds\n";
                fprintf(fp, "Run time for %s: %d hrs %d minutes %d seconds\n",options
                        ->getSimulationName(),hrs,minutes,seconds);
                fclose(fp);
        }
        delete sos;
        delete g;
        delete options;
        Material::deleteMaterials();
        return SUCCESS;
}
```

FDTDSolver.h

```
#ifndef FDTD_SOLVER_H
#define FDTD_SOLVER_H

#include "Geometry.h"
#include "Options.h"

class FDTDSolver
{
protected:
        Geometry *geometry;
        Options *options;

public:
        FDTDSolver( Geometry *geom, Options *opt );
        virtual int solve() = 0;
        virtual ~FDTDSolver();
        virtual int initialize();
};
#endif
```

FDTDSolver.cpp

```
#include "FDTDSolver.h"
#include "Geometry.h"
#include "Options.h"
#include <iostream.h>

FDTDSolver::FDTDSolver( Geometry *geom, Options* opt ){
        cout << "Creating: FDTDSolver\n";
        geometry = geom;
        options = opt;
}
FDTDSolver::~FDTDSolver(){
        cout << "Deleting: FDTDSolver\n";
        geometry = NULL;
}
int FDTDSolver::initialize(){
        int rc = Material::initializeMaterials( geometry );
```

```
        cerr << "Message: Initializing finite difference time domain solver\n";
        return rc;
}
```

GaussianSource.h

```
#ifndef GAUSSIAN_SOURCE_H
#define GAUSSIAN_SOURCE_H

#include "Material.h"
#include "Source.h"

class GaussianSource : public Source
{
protected:
        double m_dFullWidthHalfMax;
        double m_dSigma;
        int ramp;

public:
        GaussianSource( long index, char *name, Material *lastMaterial, double maxEField, int startTime, double lambda, double FWHM);
        inline double getFWHM(){return m_dFullWidthHalfMax;}
        virtual ~GaussianSource();
};

#endif
```

GaussianSource.cpp

```
#include "GaussianSource.h"
#include <iostream.h>
#include <math.h>
#include "stdheader.h"
#include "Geometry.h"
#include <stdio.h>

GaussianSource::GaussianSource( long index, char *name, Material *lastMaterial, double maxEField, int startTime, double lambda, double FWHM )
        :Source( index, name, lastMaterial, maxEField, startTime, lambda ){
        m_dFullWidthHalfMax = FWHM;
        m_dSigma = FWHM / ( 2 * sqrt( 2 * log(2) ) );
        ramp = 100;
        cerr << "Message: Creating gaussian source with lambda= "<< lambda<< "\n";
}

GaussianSource::~GaussianSource()
{
        cerr << "Message: Deleting gaussian source\n";
}
```

Geometry.h

```
#ifndef GEOMETRY_H
#define GEOMETRY_H

#include "GeometryPoint.h"
#include "Material.h"
#include "GaussianSource.h"

class Geometry
{

        //member variables
private:
        double m_dDelta;
        double m_dDeltaTime;
        double m_dMinimumLambda;
        int m_iTime;
        int m_iWidth;
        int m_iHeight;
        int m_horizontal_source_location;
        GeometryPoint **points;

        //member functions
public:
        Geometry();
        inline double getDeltaSpace() { return m_dDelta; }
        inline double getDeltaTime() { return m_dDeltaTime; }
        inline double getRunningTime() { return ( (double) m_iTime ) * m_dDeltaTime; }
        inline double getMinimumLambda() { return m_dMinimumLambda; }
        inline int getGridWidth() { return m_iWidth; }
        inline int getGridHeight() { return m_iHeight; }
        inline int getGridTime() {  return m_iTime; }
        void setHorizontalSourceLocation(int y){m_horizontal_source_location=y;}
        int getHorizontalSourceLocation(){return m_horizontal_source_location;}
        inline GeometryPoint** getPoints(){ return points; }
        inline Material* getMaterial( int i, int j) { return points[j][i].getMaterial(); }
        Source* isPointLeftOfSource( int i, int j);
        Source* isPointBelowSource( int i, int j);
```

```
        int setMinimumLambda( double lambda);

        ~Geometry();

        int setDeltaSpace( double delta );
        int setDeltaTime( double deltaTime );
        int setGridTime( int time );

        friend class BitmapGeometryReader;
        friend class BitmapSourceReader;
        friend class BitmapBoundaryConditionReader;
        friend class PropertyReader;

};

#endif
```

Geometry.cpp

```
#include "Geometry.h"
#include "GeometryPoint.h"
#include <cstdio>
#include <iostream.h>

Geometry::Geometry(){
        _points = NULL;
        m_dDelta = 0;
        m_dDeltaTime = 0;
        m_dMinimumLambda = 1E80;
        m_iWidth = 0;
        m_iHeight = 0;
}

int Geometry::setDeltaSpace( double delta ){
        m_dDelta = delta;
        return SUCCESS;
}

int Geometry::setDeltaTime( double deltaTime ){
        m_dDeltaTime = deltaTime;
        return SUCCESS;
}

int Geometry::setGridTime( int time ){
        m_iTime = time;
        return SUCCESS;
}

int Geometry::setMinimumLambda( double lambda){
        if( m_dMinimumLambda > lambda )
                m_dMinimumLambda = lambda;
        return SUCCESS;
}

Geometry::~Geometry(){
        if( points != NULL ){
                for(int i=0;i<m_iHeight;i++){
                        if( points[i] != NULL )
                                delete [] points[i];
                }
                delete [] points;
                cerr << "Deleting: geometry points\n";
        }
}

Source* Geometry::isPointLeftOfSource( int i, int j){
        if( i >= (m_iWidth-1) || j >= m_iHeight )
                return NULL;
        else if( getMaterial(i, j)->isBoundary() )
                return NULL;
        else if( getMaterial(i, j)->isSource() )
                return NULL;
        else if( getMaterial(i+1, j)->isSource() )
                return (GaussianSource*)getMaterial(i+1,j);
        else
                return NULL;
}

Source* Geometry::isPointBelowSource( int i, int j){
        if( i >= m_iWidth || j >= (m_iHeight-1) )
                return NULL;
        else if( getMaterial(i, j)->isBoundary() )
                return NULL;
        else if( getMaterial(i, j)->isSource() )
                return NULL;
        else if( getMaterial(i, j+1)->isSource() )
                return (GaussianSource*)getMaterial(i,j+1);
        else
                return NULL;
}
```

GeometryPoint.h

```
#ifndef GEOMETRY_POINT_H
#define GEOMETRY_POINT_H

#include "Material.h"

class GeometryPoint
{
        //member variables
private:
        Material *material;

private:
        void setMaterial( Material *mat );
        friend class BitmapGeometryReader;
        friend class PropertyReader;

        //member functions
public:
        GeometryPoint();
        inline Material* getMaterial() { return material; }
};

#endif
```

GeometryPoint.cpp

```
#include "GeometryPoint.h"
#include <assert.h>
#include <string.h>

GeometryPoint::GeometryPoint(){
        material = NULL;
}


void GeometryPoint::setMaterial( Material *mat ){
        assert( mat != NULL );
        material = mat;
}
```

IntensityDependentLookUp.h

```
#ifndef INTENSITYDEPENDENTLOOKUP_H
#define INTENSITYDEPENDENTLOOKUP_H

class IntensityDependentLookUp
{
public:
        IntensityDependentLookUp();
        virtual ~IntensityDependentLookUp();
        void Initialize_Arrays();
        void Set_Array_Length(int length);
        double Get_Current(double intensity_value);
private:
        double * x_values;
        double * y_values;
        int array_length;

};

#endif
```

IntensityDependentLookUp.cpp

```
#include "IntensityDependentLookUp.h"
#include <stdlib.h>
#include <fstream.h>
#include <math.h>

IntensityDependentLookUp::IntensityDependentLookUp(){
        array_length=0;
        x_values=NULL;
        y_values=NULL;
}

IntensityDependentLookUp::~IntensityDependentLookUp(){
        delete [] x_values;
        delete [] y_values;
}

void IntensityDependentLookUp::Initialize_Arrays(){
        x_values=new double[array_length];
        y_values=new double[array_length];
        ifstream input_data_x,input_data_y;
```

```
        input_data_x.open("silver_spline_data_x.txt",ios::in|ios::nocreate);
        input_data_y.open("silver_spline_data_y.txt",ios::in|ios::nocreate);

        if(!input_data_x.is_open()){
                cout<<"silver_spline_data_x.txt not found"<<endl;
                char a;
                cin>>a;
        }
        if(!input_data_y.is_open()){
                cout<<"silver_spline_data_y.txt not found"<<endl;
                char a;
                cin>>a;
        }

        for(int i=0;i<array_length;i++){
                input_data_x>>x_values[i];
                input_data_y>>y_values[i];
        }
}

double IntensityDependentLookUp::Get_Current(double intensity_value){
        if((intensity_value<(x_values[0]))){
                return 19.0993941717497*pow(intensity_value,3.08668023397964);
        }
        else{
                int location=array_length-1;
                for(int i=0;i<array_length-1;i++){
                        if((intensity_value>=x_values[i])&&(intensity_value<x_values[i+1])){
                                location=i;
                        }
                }
                if(location==array_length-1){
                        return y_values[array_length-1];
                }
                return (y_values[location+1]-y_values[location])/(x_values[location+1]
                        -x_values[location])*(intensity_value
                        -x_values[location])+y_values[location];
        }
}

void IntensityDependentLookUp::Set_Array_Length(int length){
        array_length=length;
}
```

Material.h

```
#ifndef MATERIAL_H
#define MATERIAL_H

#include "stdheader.h"
#include <math.h>


class Geometry;

class Material
{
        //static variables
private:
        static Material* firstMaterial;
        static Material* lastMaterial;

public:
        inline static Material* getMaterials(){ return firstMaterial; };
        static Material* materialExists( long index );
        static long getIndexForColor( BYTE_1 blue, BYTE_1 green, BYTE_1 red );
        static void deleteMaterials();
        static int initializeMaterials( Geometry *geometry );

        //member variables
protected:
        char* m_sMaterialName;
        long m_lIndex;
        bool m_bIsBoundary;
        bool m_bIsSource;
        bool m_bIsDispersive;
        bool m_bIsMetal;
        bool m_bIsConductive;

        //non-dispersive material constants
        double m_dMagneticConductivity;
        double m_dMagneticPermeability;
        double m_dElectricConductivity;
        double m_dElectricPermittivity;

        double m_dElectricCoefficientA;
        double m_dElectricCoefficientB;
        double m_dMagneticCoefficientA;
        double m_dMagneticCoefficientB;
```

```
        //dispersive metal constants
        double m_dWp; //plasma frequency of the metal
        double m_dSF; //scattering frequency of the metal
                    == (epsilonNot*plasmaFrequency^2)/conductivity;
        double m_dH1;
        double m_dH2;
        double m_dH3;
        double m_dH4;

public:
        Material* nextMaterial;
        Material* prevMaterial;

        //member functions

        Material( long index, char* name, double EC, double MC, double EP, double MP, double b1, double b2, double b3,
                                double c1, double c2, double c3, double d1, double d2, double d3);

        Material( long index, char* name, double EC, double MC, double EP, double MP, double wp );
        Material( long index, char* name, double EC, double MC, double EP, double MP );
        Material( long index, char* name, Material *m );
        Material( long index, char* name, double EC, double MC, double EP, double MP, double wp, double sf );
        inline bool isBoundary() { return m_bIsBoundary; }
        inline bool isSource() { return m_bIsSource; }
        inline bool isDispersive() { return m_bIsDispersive; }
        inline bool isMetal(){ return m_bIsMetal; }
        inline bool isConductive() { return m_bIsConductive; }
        inline char* getName() { return m_sMaterialName; }

        inline double getRefractiveIndex(){ return sqrt(m_dElectricPermittivity/EPSILON_NOT); }

        inline double getElectricConductivity(){ return m_dElectricConductivity; }
        inline double getMagneticConductivity(){ return m_dMagneticConductivity; }
        inline double getElectricPermittivity(){ return m_dElectricPermittivity; }
        inline double getMagneticPermeability(){ return m_dMagneticPermeability; }

        inline double getElectricCoefficientA(){ return m_dElectricCoefficientA; };
        inline double getElectricCoefficientB(){ return m_dElectricCoefficientB; };
        inline double getMagneticCoefficientA(){ return m_dMagneticCoefficientA; };
        inline double getMagneticCoefficientB(){ return m_dMagneticCoefficientB; };

        inline double getCoefficientH1(){ return m_dH1; }
        inline double getCoefficientH2(){ return m_dH2; }
        inline double getCoefficientH3(){ return m_dH3; }
        inline double getCoefficientH4(){ return m_dH4; }
        void setElectricPermittivity(double perm){m_dElectricPermittivity=perm;}

        inline double getPlasmaFrequency(){ return m_dWp; }
        inline double getScatteringFrequency(){ return m_dSF; }

        inline long getIndex(){ return m_lIndex; }
        virtual ~Material();

        friend class PropertyReader;

protected:
        virtual int initialize( Geometry *geometry );
};

#endif
```

Material.cpp

```
#include <string.h>
#include <iostream.h>
#include "Material.h"
#include "Geometry.h"

Material* Material::firstMaterial=NULL;
Material* Material::lastMaterial=NULL;

//member functions
Material::Material( long index, char* name, double EC, double MC, double EP, double MP, double b1, double b2, double b3, double c1, double c2,
        double c3, double d1, double d2, double d3 ){
        m_lIndex = index;
        m_bIsBoundary = false;
        m_bIsSource = false;
        m_bIsDispersive = true;
        m_bIsMetal = false;
        m_dElectricConductivity = EC;
        m_dMagneticConductivity = MC;
        m_dElectricPermittivity = EP * EPSILON_NOT;
        m_dMagneticPermeability = MP * MU_NOT;
        m_dWp = 0;
        m_dSF = 0;
        m_sMaterialName = new char[strlen(name)+1];
        strcpy(m_sMaterialName, name);
        nextMaterial = NULL;
        prevMaterial = NULL;

        //add material to list
```

```
        if( firstMaterial == NULL ){
                firstMaterial = lastMaterial = this;
        }
        else{
                lastMaterial->nextMaterial = this;
                this->prevMaterial = lastMaterial;
                lastMaterial = this;
        }
        cerr << "Creating: " << m_sMaterialName << "\n";
}

Material::Material( long index, char* name, double EC, double MC, double EP, double MP, double wp ){
        m_lIndex = index;
        m_bIsBoundary = false;
        m_bIsSource = false;
        m_bIsMetal = true;
        m_bIsDispersive = true;
        m_dElectricConductivity = EC;
        m_dMagneticConductivity = MC;
        m_dElectricPermittivity = EP * EPSILON_NOT;
        m_dMagneticPermeability = MP * MU_NOT;
        m_dWp = wp;
        m_dSF = m_dElectricConductivity;
        m_sMaterialName = new char[strlen(name)+1];
        strcpy(m_sMaterialName, name);
        nextMaterial = NULL;
        prevMaterial = NULL;

        //add material to list
        if( firstMaterial == NULL ){
                firstMaterial = lastMaterial = this;
        }
        else{
                lastMaterial->nextMaterial = this;
                this->prevMaterial = lastMaterial;
                lastMaterial = this;
        }
        cerr << "Creating: " << m_sMaterialName << "\n";
}


Material::Material( long index, char* name, double EC, double MC, double EP, double MP, double wp, double sf ){
        m_lIndex = index;
        m_bIsBoundary = false;
        m_bIsSource = false;
        m_bIsMetal = false;
        m_bIsConductive = true;
        m_bIsDispersive = true;
        m_dElectricConductivity = EC;
        m_dMagneticConductivity = MC;
        m_dElectricPermittivity = EP * EPSILON_NOT;
        m_dMagneticPermeability = MP * MU_NOT;
        m_dWp = wp;
        m_dSF = sf;
        m_sMaterialName = new char[strlen(name)+1];
        strcpy(m_sMaterialName, name);
        nextMaterial = NULL;
        prevMaterial = NULL;

        //add material to list
        if( firstMaterial == NULL ){
                firstMaterial = lastMaterial = this;
        }
        else{
                lastMaterial->nextMaterial = this;
                this->prevMaterial = lastMaterial;
                lastMaterial = this;
        }
        cerr << "Creating: " << m_sMaterialName << "\n";
}

Material::Material( long index, char* name, double EC, double MC, double EP, double MP ){
        m_lIndex = index;
        m_bIsBoundary = false;
        m_bIsSource = false;
        m_bIsDispersive = false;
        m_bIsMetal = false;
        m_dElectricConductivity = EC;
        m_dMagneticConductivity = MC;
        m_dElectricPermittivity = EP * EPSILON_NOT;
        m_dMagneticPermeability = MP * MU_NOT;
        m_sMaterialName = new char[strlen(name)+1];
        strcpy(m_sMaterialName, name);
        nextMaterial = NULL;
        prevMaterial = NULL;

        //add material to list
        if( firstMaterial == NULL ){
                firstMaterial = lastMaterial = this;
        }
        else{
                lastMaterial->nextMaterial = this;
                this->prevMaterial = lastMaterial;
```

```
                lastMaterial = this;
        }
        cerr << "Creating: " << m_sMaterialName << "\n";
}

Material::Material( long index, char* name, Material *m ){
        m_lIndex = index;
        m_bIsBoundary = false;
        m_bIsSource = false;
        m_dElectricConductivity = m->getElectricConductivity();
        m_dMagneticConductivity = m->getMagneticConductivity();
        m_dElectricPermittivity = m->getElectricPermittivity();
        m_dMagneticPermeability = m->getMagneticPermeability();
        m_dWp = m->m_dWp;
        m_dSF = m->m_dSF;
        m_bIsDispersive = m->m_bIsDispersive;
        m_bIsMetal = m->m_bIsMetal;
        m_sMaterialName = new char[strlen(name)+1];
        strcpy(m_sMaterialName, name);
        nextMaterial = NULL;
        prevMaterial = NULL;

        //add material to list
        if( firstMaterial == NULL ){
                firstMaterial = lastMaterial = this;
        }
        else{
                lastMaterial->nextMaterial = this;
                this->prevMaterial = lastMaterial;
                lastMaterial = this;
        }
        cerr << "Message: Creating " << m_sMaterialName << "\n";
}

Material::~Material(){
        if( this->prevMaterial != NULL ){
                this->prevMaterial->nextMaterial = this->nextMaterial;
        }
        if( this->nextMaterial != NULL )
        this->nextMaterial->prevMaterial = this->prevMaterial;

        cerr << "Deleting: " << m_sMaterialName << "\n";
        delete m_sMaterialName;
        m_sMaterialName = NULL;
}

Material* Material::materialExists( long index ){
        Material *mat = firstMaterial;

        while( mat != NULL )        {
                if( mat->getIndex() == index )
                        return mat;
                mat=mat->nextMaterial;
        }
        cerr << "Checking index: " << index << "\n";
        mat = firstMaterial;
        while( mat != NULL )        {
                cerr << "Index is: " << mat->getIndex() << " for material "
                        << mat->getName() << "\n";
                if( mat->getIndex() == index )
                        return mat;
                mat=mat->nextMaterial;
        }
        return NULL;
}

long Material::getIndexForColor( BYTE_1 blue, BYTE_1 green, BYTE_1 red ){
        long index = 0;
        index |= blue<<16;
        index |= green<<8;
        index |= red;
        return index;
}

int Material::initializeMaterials( Geometry *geometry ){
        Material *mat = firstMaterial;
        int rc = SUCCESS;

        while( mat != NULL ){
                rc |= mat->initialize( geometry );
                mat = mat->nextMaterial;
        }

        return rc;
}

int Material::initialize( Geometry *geometry ){
        int rc = SUCCESS;
        double deltaTime = geometry->getDeltaTime();
        double deltaSpace = geometry->getDeltaSpace();
        cerr << "Message: Initializing material " << this->getName() << "\n";

        m_dElectricCoefficientA=( 1 - m_dElectricConductivity*deltaTime/( 2*m_dElectricPermittivity ) )
```

```
                        / (1+m_dElectricConductivity*deltaTime/( 2*m_dElectricPermittivity));
        m_dElectricCoefficientB= ( deltaTime/( deltaSpace*m_dElectricPermittivity))
                        / ( 1+m_dElectricConductivity*deltaTime/( 2*m_dElectricPermittivity));
        m_dMagneticCoefficientA=( 1 - m_dMagneticConductivity*deltaTime/( 2*m_dMagneticPermeability ) )
                        / ( 1 + m_dMagneticConductivity*deltaTime/(2*m_dMagneticPermeability)) ;
        m_dMagneticCoefficientB (deltaTime/( deltaSpace*m_dMagneticPermeability ) )
                        / ( 1 + m_dMagneticConductivity*deltaTime/( 2*m_dMagneticPermeability ) ) ;


        m_dH1 = (2.0 + getScatteringFrequency()*geometry->getDeltaTime() );
        m_dH2 = (2.0 - getScatteringFrequency()*geometry->getDeltaTime() );
        m_dH3 = ( getPlasmaFrequency()*getPlasmaFrequency()*geometry->getDeltaTime()*geometry->getDeltaTime()*EPSILON_NOT
                - getScatteringFrequency()*geometry->getDeltaTime()*getElectricPermittivity() + 2*getElectricPermittivity() );
        m_dH4 = ( getPlasmaFrequency()*getPlasmaFrequency()*geometry->getDeltaTime()*geometry->getDeltaTime()*EPSILON_NOT
                + getScatteringFrequency()*geometry->getDeltaTime()*getElectricPermittivity() + 2*getElectricPermittivity() );

        return rc;
}

void Material::deleteMaterials(){
        Material *mat = firstMaterial;
        while( mat != NULL )    {
                Material *nextMat = mat->nextMaterial;
                delete mat;
                mat = nextMat;
        }
        firstMaterial = lastMaterial = NULL;
}
```

Options.h

```
#ifndef OPTIONS_H
#define OPTIONS_H

class Options
{
protected:
        int m_iFrameSpacing;
        char *m_sSimulationName;
        int m_iSimulationMode;
        bool m_bPulsedMode;
        double m_dPulseLength;
        int m_iOutputMode;
        double m_dMaxOutputFieldStrength;
        double m_dMinOutputFieldStrength;
        int *m_IXSamplingPoints;
        int *m_IYSamplingPoints;
        int m_iSamplingPointCount;
        int m_iXIntegrationLineCount;
        int m_iYIntegrationLineCount;
        int *m_IXIntegrationLine;
        int *m_IYIntegrationLine;

        bool m_bQueryContinue;

        void setPulsedMode( bool mode );
        void setFrameSpacing( int spacing );
        void setSimulationName( char* simulationName );
        void setSimulationMode( char* mode );
        void setOutputMode( char* mode, double min, double max );
        void setTimeDomainFieldSamplingPointCount( int count );
        void setTimeDomainFieldSamplingPoint( int index, int x, int y );
        void setXIntegrationLine( int index, int x );
        void setXIntegrationLineCount( int count );
        void setYIntegrationLine( int index, int y );
        void setYIntegrationLineCount( int count );
        void setQueryContinue( bool cont );

public:
        Options();
        inline int getFrameSpacing(){ return m_iFrameSpacing; }
        inline int isQueryContinue(){ return m_bQueryContinue; }
        inline char* getSimulationName(){ return m_sSimulationName; }
        inline int getSimulationMode(){ return m_iSimulationMode; }
        inline bool isPulsedMode(){ return m_bPulsedMode; }
        inline double getMinOutputField(){ return m_dMinOutputFieldStrength; }
        inline double getMaxOutputField(){ return m_dMaxOutputFieldStrength; }
        inline int getOutputMode(){ return m_iOutputMode; }
        inline int getSamplingPointCount(){ return m_iSamplingPointCount; }
        inline int getXSamplingPoint( int index ){ return m_IXSamplingPoints[index]; }
        inline int getYSamplingPoint( int index ){ return m_IYSamplingPoints[index]; }
        inline int getXIntegrationLine( int index ){ return m_IXIntegrationLine[index]; }
        inline int getXIntegrationLineCount(){ return m_iXIntegrationLineCount; }
        inline int getYIntegrationLine( int index ){ return m_IYIntegrationLine[index]; }
        inline int getYIntegrationLineCount(){ return m_iYIntegrationLineCount; }
        ~Options();
        friend class PropertyReader;
        friend class SecondOrderSolver;
};

#endif
```

Options.cpp

```
#include "Options.h"
#include "stdheader.h"
#include <iostream.h>
#include <math.h>
#include <string.h>

Options::Options(){
		m_iFrameSpacing = 0;
		m_sSimulationName = NULL;
		m_bPulsedMode = false;
		m_iXIntegrationLineCount = 0;
		m_iYIntegrationLineCount = 0;
		cerr << "Message: Creating options object\n";
}

Options::~Options(){
		cerr << "Message: Deleting options object\n";
		delete [] m_sSimulationName;
		if( m_IXSamplingPoints != NULL )
			delete [] m_IXSamplingPoints;
		if( m_IYSamplingPoints != NULL )
			delete [] m_IYSamplingPoints;
		m_iFrameSpacing = 0;
}

void Options::setFrameSpacing( int spacing ){
		m_iFrameSpacing = abs(spacing);
}

void Options::setSimulationName( char* simulationName ){
		if( simulationName == NULL ){
			m_sSimulationName = new char[strlen("fdtd")+1];
			strcpy(m_sSimulationName,"fdtd");
		}
		else if( strlen(simulationName) == 0 ){
			m_sSimulationName = new char[strlen("fdtd")+1];
			strcpy(m_sSimulationName,"fdtd");
		}
		else{
			m_sSimulationName = new char[strlen(simulationName)+1];
			strcpy(m_sSimulationName, simulationName);
		}
}

void Options::setSimulationMode( char* mode ){
		//cerr << "Simulation mode " << mode;
		if( strcmp(mode, TM_TE_MODE_TOKEN) == 0 )
			m_iSimulationMode = TM_TE_MODE;
		else if( strcmp(mode, TM_MODE_TOKEN) == 0 )
			m_iSimulationMode = TM_MODE;
		else if( strcmp(mode, TE_MODE_TOKEN) == 0 )
			m_iSimulationMode = TE_MODE;
		else
			m_iSimulationMode = TM_TE_MODE;
}

void Options::setPulsedMode( bool mode ){
		m_bPulsedMode = mode;
}

void Options::setOutputMode( char* mode, double min, double max ){
		//cerr << "Output mode " << mode;
		if( strcmp(mode, BMP_TXT_OUTPUT_MODE_TOKEN) == 0 )
			m_iOutputMode = BMP_TXT_OUTPUT_MODE;
		else if( strcmp(mode, BMP_OUTPUT_MODE_TOKEN) == 0 )
			m_iOutputMode = BMP_OUTPUT_MODE;
		else if( strcmp(mode, TXT_OUTPUT_MODE_TOKEN) == 0 )
			m_iOutputMode = TXT_OUTPUT_MODE;
		else
			m_iOutputMode = BMP_OUTPUT_MODE;
		m_dMaxOutputFieldStrength = max;
		m_dMinOutputFieldStrength = min;
}

void Options::setTimeDomainFieldSamplingPointCount( int count ){
		m_iSamplingPointCount = count;
		if( count > 0 ){
			m_IXSamplingPoints = new int[count];
			m_IYSamplingPoints = new int[count];
		}
		else{
			m_IXSamplingPoints = NULL;
			m_IYSamplingPoints = NULL;
		}
}

void Options::setTimeDomainFieldSamplingPoint( int index, int x, int y ){
		m_IXSamplingPoints[index] = x;
		m_IYSamplingPoints[index] = y;
}
```

```
void Options::setXIntegrationLineCount( int count ){
		m_iXIntegrationLineCount = count;
		if( count > 0 ){
			m_IXIntegrationLine = new int[count];
		}
		else{
			m_IXIntegrationLine = NULL;
		}
}

void Options::setYIntegrationLineCount( int count ){
		m_iYIntegrationLineCount = count;
		if( count > 0 ){
			m_IYIntegrationLine = new int[count];
		}
		else{
			m_IYIntegrationLine = NULL;
		}
}

void Options::setXIntegrationLine( int index, int x ){
		m_IXIntegrationLine[index] = x;
}

void Options::setYIntegrationLine( int index, int y ){
		m_IYIntegrationLine[index] = y;
}

void Options::setQueryContinue( bool cont ){
		m_bQueryContinue = cont;
}
```

PMLBC.h

```
#ifndef PML_BOUNDARY_CONDITION_H
#define PML_BOUNDARY_CONDITION_H

#include "stdheader.h"
#include "Material.h"
#include "iostream.h"

class Geometry;

class PMLBC : public Material
{

protected:
		static PMLBC* boundaryCondition;

public:
		static inline PMLBC*
			getBoundaryCondition(){ return boundaryCondition; }

		//member variables
private:
		int m_iWidth;
		double* m_DElectricExpCoefficientsA;
		double* m_DMagneticExpCoefficientsA;
		double* m_DElectricExpCoefficientsB;
		double* m_DMagneticExpCoefficientsB;

		//member functions
public:
		PMLBC( long index, char* name, int width, double ec );
		inline int getWidth(){ return m_iWidth; }
		double getElectricFieldExponentialCoefficientA(Geometry *geometry, double current_epsilon, int pos);
		double getMagneticFieldExponentialCoefficientA(Geometry *geometry, double current_epsilon, int pos);
		double getElectricFieldExponentialCoefficientB(Geometry *geometry, double current_epsilon, int pos);
		double getMagneticFieldExponentialCoefficientB(Geometry *geometry, double current_epsilon, int pos);
		double getElectricCoefficientA(Geometry *geometry, double current_epsilon);
		double getElectricCoefficientB(Geometry *geometry, double current_epsilon);
		double getMagneticCoefficientA(Geometry *geometry, double current_epsilon);
		double getMagneticCoefficientB(Geometry *geometry, double current_epsilon);
		virtual int initialize( Geometry *geometry );
		virtual ~PMLBC();
};

#endif
```

PMLBC.cpp

```
#include <string.h>
#include <iostream.h>
#include "PMLBC.h"
#include "stdheader.h"
#include "Geometry.h"
#include <math.h>
```

```
PMLBC* PMLBC::boundaryCondition = NULL;

//member functions
PMLBC::PMLBC( long index, char* name, int width, double ec )
:Material( index, name, 0, 0, 1.0, 1.0 ){
        cerr << "Creating: Perfectly matched boundary layer with conductivity " << ec<< " and width " << width << "\n";
        m_dElectricConductivity = ec;
        m_dMagneticConductivity = m_dElectricConductivity * MU_NOT ;
        m_iWidth = width;
        m_blsBoundary = true;
        boundaryCondition = this;
        m_DMagneticExpCoefficientsA = new double[m_iWidth];
        m_DElectricExpCoefficientsA = new double[m_iWidth];
        m_DMagneticExpCoefficientsB = new double[m_iWidth];
        m_DElectricExpCoefficientsB = new double[m_iWidth];
}

int PMLBC::initialize( Geometry *geometry ){
        Material::initialize( geometry );
        int rc = SUCCESS;
        cerr << "Message: Initializing boundary condition " << this->getName() << "\n";
        double deltaTime = geometry->getDeltaTime();
        double deltaSpace = geometry->getDeltaSpace();

        return rc;

}


double PMLBC::getElectricFieldExponentialCoefficientA(Geometry *geometry,double current_epsilon, int pos){
        double sigmaE = m_dElectricConductivity*(double)(pos+0.5)/(double)m_iWidth;
        return exp(-sigmaE*geometry->getDeltaTime()/current_epsilon);;
}

double PMLBC::getMagneticFieldExponentialCoefficientA(Geometry *geometry, double current_epsilon, int pos){
        double sigmaM = m_dMagneticConductivity*(double)pos/(double)m_iWidth/current_epsilon;
        return exp(-sigmaM*geometry->getDeltaTime()/MU_NOT);
}

double PMLBC::getElectricFieldExponentialCoefficientB(Geometry *geometry, double current_epsilon, int pos){
        double sigmaE = m_dElectricConductivity*(double)(pos+0.5)/(double)m_iWidth;
        return (1-exp(-sigmaE*geometry->getDeltaTime()/current_epsilon))/sigmaE/geometry->getDeltaSpace();
}

double PMLBC::getMagneticFieldExponentialCoefficientB(Geometry *geometry, double current_epsilon, int pos){
        double sigmaM = m_dMagneticConductivity*(double)pos/(double)m_iWidth/current_epsilon;
        if(pos==0){
                return geometry->getDeltaTime()*INVERSE_MU_NOT/geometry->getDeltaSpace();
        }
        else{
                return (1-exp(-sigmaM*geometry->getDeltaTime()/MU_NOT))/sigmaM/geometry->getDeltaSpace();
        }
}

double PMLBC::getElectricCoefficientA(Geometry *geometry, double current_epsilon){
        return ( 1 - m_dElectricConductivity*geometry->getDeltaTime()/( 2*current_epsilon ) ) /(1+m_dElectricConductivity
                *geometry->getDeltaTime()/( 2*current_epsilon ) );
}
double PMLBC::getElectricCoefficientB(Geometry *geometry, double current_epsilon){
        return ( geometry->getDeltaTime()/( geometry->getDeltaSpace()*current_epsilon ) ) / ( 1 + m_dElectricConductivity
                *geometry->getDeltaTime()/( 2*current_epsilon ) );
}
double PMLBC::getMagneticCoefficientA(Geometry *geometry,
        double current_epsilon){
        double sigmaM = m_dMagneticConductivity/current_epsilon;
        return ( 1 - sigmaM*geometry->getDeltaTime()/( 2*MU_NOT ) ) / ( 1 + sigmaM*geometry->getDeltaTime()/( 2*MU_NOT ) );
}
double PMLBC::getMagneticCoefficientB(Geometry *geometry,
        double current_epsilon){
        double sigmaM = m_dMagneticConductivity/current_epsilon;
        return ( geometry->getDeltaTime()/( geometry->getDeltaSpace()*MU_NOT ) ) / ( 1 + sigmaM*geometry->getDeltaTime()/( 2*MU_NOT ) );
}

PMLBC::~PMLBC(){
        if( m_DElectricExpCoefficientsA != NULL )
                delete m_DElectricExpCoefficientsA;
        if( m_DMagneticExpCoefficientsA != NULL )
                delete m_DMagneticExpCoefficientsA;
        if( m_DElectricExpCoefficientsB != NULL )
                delete m_DElectricExpCoefficientsB;
        if( m_DMagneticExpCoefficientsB != NULL )
                delete m_DMagneticExpCoefficientsB;
}
```

PropertyReader.h

```
#ifndef PROPERTY_READER_H
#define PROPERTY_READER_H
```

```
#include "Geometry.h"
#include "Options.h"

class PropertyReader
{

        //member variables
private:
        Geometry *geometry;
        Options *options;
        int readState;
        Material* lastMaterial;

public:
        PropertyReader();
        int readProperties( char *propfile, Geometry* geom, Options* options);
        ~PropertyReader();

private:
        int parseLine( char* line );
        int parseMaterial( char* line );
        int parseMaterialType(long index, char* name, char* type, char *line);
        int parseGrid( char* line );
        int parseOptions( char* line );

};

#endif
```

PropertyReader.cpp

```
#include "PropertyReader.h"
#include <cstdio>
#include <cstring>
#include <iostream.h>
#include "Geometry.h"
#include <assert.h>
#include "Material.h"
#include <math.h>
#include "GaussianSource.h"
#include "PulsedGaussianSource.h"
#include "PMLBC.h"

PropertyReader::PropertyReader(){
        readState = -1;
        geometry = NULL;
        cerr << "Creating: PropertyReader\n";
}

/** Read the property file */
int PropertyReader::readProperties( char *propFile, Geometry *geom, Options *opt ){
        FILE* propertyFP;
        int rc = SUCCESS;
        char line[2048];
        geometry = geom;
        options = opt;
        lastMaterial = NULL;

        propertyFP = fopen( propFile, "rt");
        if( propertyFP == NULL ){
                cerr << "Property file not found!\n";
                return FILE_NOT_FOUND;
        }

        while( (fgets(line, 2048, propertyFP ) != NULL) && (rc == SUCCESS) ){
                rc = parseLine( line );
        }

        fclose( propertyFP );
        return rc;
}

int PropertyReader::parseLine( char* line ){
        assert( line != NULL );
        int rc = SUCCESS;
        if( strstr(line, GRID_TOKEN ) != NULL )
                readState = GRID_STATE;
        else if( strstr(line, MATERIAL_TOKEN ) != NULL )
                readState = MATERIAL_STATE;
        else if( strstr(line, OPTIONS_TOKEN ) != NULL )
                readState = OPTIONS_STATE;
        else if( line[0] == '%' )
                cerr << "Comment: " << &line[1];
        else if( readState == MATERIAL_STATE )
                rc = parseMaterial( line );
        else if( readState == GRID_STATE )
                rc = parseGrid( line );
        else if( readState == OPTIONS_STATE )
                rc = parseOptions( line );
        else
                cerr << "Warning: Invalid property! Skipping line...\n";
```

```
		return rc;
}

int PropertyReader::parseGrid( char* line ){
		int rc = SUCCESS;
		char *position;
		if( (position = strstr(line, DELTA_TOKEN)) != NULL ){
				double delta=0;
				position = position + strlen(DELTA_TOKEN);
				if( sscanf( position,"%le",&delta ) != 1 )
						rc = INVALID_FILE_FORMAT;
				geometry->setDeltaSpace( delta );
		}
		else if( (position = strstr(line, DELTA_TIME_TOKEN)) != NULL ){
				double time;
				position = position + strlen(DELTA_TIME_TOKEN);
				if( sscanf( position,"%le",&time ) != 1 )
						rc = INVALID_FILE_FORMAT;
				geometry->setDeltaTime( time );
		}
		else if( (position = strstr(line, GRID_TIME_TOKEN)) != NULL ){
				int time;
				position = position + strlen(GRID_TIME_TOKEN);
				if( sscanf( position,"%d",&time ) != 1 )
						rc = INVALID_FILE_FORMAT;
				geometry->setGridTime( time );
		}
		else{
				cerr << "Unknown token: " << line;
		}
		return rc;
}

int PropertyReader::parseMaterial( char* line ){
		int rc = SUCCESS;
		char name[2048];
		char type[2048];
		unsigned char b,g,r;
		if( sscanf(line, "%hhi %hhi %hhi %s %s",&b, &g, &r,name, type) != 5 ){
				rc = INVALID_FILE_FORMAT;
		}
		else{
				long index = Material::getIndexForColor(b,g,r);
				Material *material = Material::materialExists( index );
				if( material == NULL ){
						line = strstr( line, type );
						line += strlen( type );
						rc = parseMaterialType(index, name, type, line);
				}
				else{
						cerr << "Error: Attempted to read in an existing material";
						rc = INVALID_FILE_FORMAT;
				}
		}
		return rc;
}

int PropertyReader::parseMaterialType(long index, char* name, char* type, char *line){
		int rc = SUCCESS;
		Material *mat = NULL;

		if( strcmp( type, DISPERSIVE_MATERIAL_TYPE_TOKEN ) == 0 ){
				double ec, mc, ep, mp;
				double b1,b2,b3,c1,c2,c3,d1,d2,d3;
				if( sscanf(line, "%le %le %le %le %le %le %le %le %le %le %le %le %le %le", &ec, &mc, &ep, &mp, &b1, &b2, &b3,
						&c1, &c2, &c3, &d1, &d2, &d3 ) == 13 ){
						mat = new Material( index, name, ec, mc, ep, mp, b1, b2, b3, c1, c2, c3, d1, d2, d3 );
						this->lastMaterial = mat;
				}
				else{
						rc = INVALID_FILE_FORMAT;
				}
		}
		else if( strcmp( type, METAL_MATERIAL_TYPE_TOKEN ) == 0 ){
				double ec, mc, ep, mp;
				double wp;
				if( sscanf(line, "%le %le %le %le %le", &ec, &mc, &ep, &mp, &wp) == 5 ){
						mat = new Material( index, name, ec, mc, ep, mp, wp );
						this->lastMaterial = mat;
				}
				else{
						rc = INVALID_FILE_FORMAT;
				}
		}
		else if( strcmp( type, CONDUCTIVE_MATERIAL_TYPE_TOKEN ) == 0 ){
				double ec, mc, ep, mp;
				double wp, sf;
				if( sscanf(line, "%le %le %le %le %le %le", &ec, &mc, &ep, &mp, &wp, &sf) == 6 ){
						mat = new Material( index, name, ec, mc, ep, mp, wp, sf );
						this->lastMaterial = mat;
				}
				else{
						rc = INVALID_FILE_FORMAT;
```

```
                }
        }
        else if( strcmp( type, MATERIAL_TYPE_TOKEN ) == 0 ){
                double ec, mc, ep, mp;
                if( sscanf(line, "%le %le %le %le", &ec, &mc, &ep, &mp) == 4 ){
                        mat = new Material( index, name, ec, mc, ep, mp );
                        this->lastMaterial = mat;
                }
                else{
                        rc = INVALID_FILE_FORMAT;
                }
        }
        else if( strcmp( type, BOUNDARY_TYPE_TOKEN ) == 0 ){
                char bcType[2048];
                if( sscanf(line, "%s", bcType) != 1 ){
                        rc = INVALID_FILE_FORMAT;
                }
                else{
                        line = strstr(line, bcType) + strlen(bcType);
                        if( strcmp( bcType, PERFECTLY_MATCHED_LAYER_TOKEN ) == 0 ){
                                double ec;
                                int width;
                                if( sscanf(line, "%d %le", &width, &ec ) != 2 ){
                                        cerr << "Error: Invalid perfectly matched boundary condition parameters\n";
                                        rc = INVALID_FILE_FORMAT;
                                }
                                else{
                                        mat = new PMLBC( index, name, width, ec );
                                }
                        }
                        else{
                                cerr << "Warning: Invalid boundary type\n";
                                rc = INVALID_FILE_FORMAT;
                        }
                }
        }
        else if( strcmp( type, SOURCE_TYPE_TOKEN ) == 0 ){
                char srcType[2048];
                if( sscanf(line, "%s", srcType) != 1 ){
                        rc = INVALID_FILE_FORMAT;
                }
                else{
                        line = strstr(line,srcType) + strlen(srcType);
                        if( strcmp( srcType, GAUSSIAN_SOURCE_TOKEN ) == 0 ){
                                if( lastMaterial == NULL ){
                                        cerr << "Error: Invalid source location\n";
                                        cerr << "Error: Source definition must be placed directly after its corresponding material\n";
                                        rc = INVALID_FILE_FORMAT;
                                }
                                else{
                                        double eField, FWHM, lambda;
                                        int startTime;
                                        if( sscanf(line," %le %d %le %le", &eField, &startTime, &lambda, &FWHM) != 4 ){
                                                cerr << "Error: Invalid Source Options\n";
                                                rc = INVALID_FILE_FORMAT;
                                        }
                                        mat = new GaussianSource( index, name, lastMaterial, eField, startTime, lambda, FWHM );
                                }
                        }
                        else if( strcmp( srcType, PULSED_GAUSSIAN_SOURCE_TOKEN ) == 0 ){
                                if( lastMaterial == NULL ){
                                        cerr << "Error: Invalid source location\n";
                                        cerr << "Error: Source definition must be
                                                placed directly after its corresponding
                                                material\n";
                                        rc = INVALID_FILE_FORMAT;
                                }
                                else{
                                        double eField, FWHM, pFWHM, phi, lambda;
                                        int startTime;
                                        if( sscanf(line," %le %d %le %le %le %le", &eField, &startTime,
                                                &lambda, &FWHM, &pFWHM, &phi) != 6 ){
                                                cerr << "Error: Invalid Source Options\n";
                                                rc = INVALID_FILE_FORMAT;
                                        }
                                        mat = new PulsedGaussianSource( index, name, lastMaterial, eField,
                                                startTime, lambda, FWHM, pFWHM, phi);
                                        options->setPulsedMode( true );
                                }
                        }
                        else{
                                cerr << "Warning: Invalid source type - "
                                        << srcType <<"\n";
                                rc = INVALID_FILE_FORMAT;
                        }
                }
        }
        else{
                cerr << "Warning: Invalid material type\n";
                rc = INVALID_FILE_FORMAT;
        }
        return rc;
}
```

```
int PropertyReader::parseOptions( char* line ){
	int rc = SUCCESS;
	char *position;
	if( (position = strstr(line, FRAME_SPACING_TOKEN)) != NULL ){
		int frameSpacing=0;
		position = position + strlen(FRAME_SPACING_TOKEN);
		if( sscanf( position,"%d",&frameSpacing) != 1 )
			rc = INVALID_FILE_FORMAT;
		options->setFrameSpacing(frameSpacing);
	}
	else if( (position = strstr(line, SIMULATION_NAME_TOKEN)) != NULL ){
		char name[2048];
		position = position + strlen(SIMULATION_NAME_TOKEN);
		if( sscanf( position,"%s",&name ) != 1 )
			rc = INVALID_FILE_FORMAT;
		options->setSimulationName( name );
	}
	else if( (position = strstr(line, SIMULATION_MODE_TOKEN)) != NULL ){
		char mode[2048];
		position = position + strlen(SIMULATION_MODE_TOKEN);
		if( sscanf( position,"%s",&mode ) != 1 )
			rc = INVALID_FILE_FORMAT;
		options->setSimulationMode( mode );
	}
	else if( (position = strstr(line, SIMULATION_OUTPUT_TOKEN)) != NULL ){
		char mode[2048];
		double min, max;
		position = position + strlen(SIMULATION_OUTPUT_TOKEN);
		if( sscanf( position,"%s %le %le",&mode,&min,&max ) != 3 )
			rc = INVALID_FILE_FORMAT;
		options->setOutputMode( mode, min, max );
	}
	else if( (position = strstr(line, QUERY_CONTINUE_TOKEN)) != NULL ){

		int cont;
		position = position + strlen(QUERY_CONTINUE_TOKEN);
		if( sscanf( position,"%d",&cont ) != 1 )
			rc = INVALID_FILE_FORMAT;
		options->setQueryContinue( cont!=0 );
		if ( cont )
			cerr << "Message: Option to continue is active.\n";
		else
			cerr << "Message: Option to continue is dormant.\n";
	}
	else if( (position = strstr(line, TIME_DOMAIN_FIELD_OUTPUT_TOKEN)) != NULL ){
		int count;
		int x,y;
		char test[2048];
		position = position + strlen(TIME_DOMAIN_FIELD_OUTPUT_TOKEN);
		if( sscanf( position,"%d ", &count ) != 1 )
			rc = INVALID_FILE_FORMAT;
		options->setTimeDomainFieldSamplingPointCount( count );
		count--;
		sprintf(test,"%d ",count);
		position = position + strlen( test );
		for( ; count>=0&&rc==SUCCESS ; count-- ){
			if( sscanf( position,"%d %d ", &x, &y ) != 2 )
				rc = INVALID_FILE_FORMAT;
			options->setTimeDomainFieldSamplingPoint( count, x, y );
			sprintf(test,"%d %d ",x,y);
			position = position + strlen( test );
		}
		cerr << "Message: Sampling " << options->getSamplingPointCount()
			<< " points for output!\n";
	}
	else if( (position = strstr(line, X_FIELD_INTEGRATOR_TOKEN)) != NULL ){
		int count;
		int x;
		char test[2048];
		position = position + strlen(X_FIELD_INTEGRATOR_TOKEN);
		if( sscanf( position,"%d ", &count ) != 1 )
			rc = INVALID_FILE_FORMAT;
		options->setXIntegrationLineCount( count );
		count--;
		sprintf(test,"%d ",count);
		position = position + strlen( test );
		for( ; count>=0&&rc==SUCCESS ; count-- ){
			if( sscanf( position,"%d ", &x) != 1 )
				rc = INVALID_FILE_FORMAT;
			options->setXIntegrationLine( count, x );
			sprintf(test,"%d ",x);
			position = position + strlen( test );
		}
		cerr << "Message: Sampling " << options->getXIntegrationLineCount()
			<< " x integration lines for output!\n";
	}
	else if( (position = strstr(line, Y_FIELD_INTEGRATOR_TOKEN)) != NULL ){
		int count;
		int y;
		char test[2048];
		position = position + strlen(Y_FIELD_INTEGRATOR_TOKEN);
		if( sscanf( position,"%d ", &count ) != 1 )
```

```
                        rc = INVALID_FILE_FORMAT;
                options->setYIntegrationLineCount( count );
                count--;
                sprintf(test,"%d ",count);
                position = position + strlen( test );
                for( ; count>=0&&rc==SUCCESS ; count-- ){
                        if( sscanf( position,"%d ", &y) != 1 )
                                rc = INVALID_FILE_FORMAT;
                        options->setYIntegrationLine( count, y );
                        sprintf(test,"%d ",y);
                        position = position + strlen( test );
                }
                cerr << "Message: Sampling " << options->getYIntegrationLineCount()
                        << " y integration lines for output!\n";
        }
        else{
                cerr << "What!\n";
        }
        return rc;
}

PropertyReader::~PropertyReader(){
        cerr << "Deleting: PropertyReader\n";
}
```

PulsedGaussianSource.h

```
#ifndef PULSED_GAUSSIAN_SOURCE_H
#define PULSED_GAUSSIAN_SOURCE_H

#include "GaussianSource.h"

class PulsedGaussianSource : public GaussianSource
{
protected:
        double m_dPulseWidth;
        double m_dPhi;
        double m_dPulseSigma;
        double m_dPulseOffset;
        double m_dChirp;

public:
        PulsedGaussianSource( long index, char *name, Material *lastMaterial, double maxEField, int startTime, double lambda, double FWHM,
                        double pulseFWHM, double phi );
        inline double getPulseDuration(){return m_dPulseWidth;}
        inline double getPulseFWHM(){return m_dFullWidthHalfMax;}
        inline double getMaxElectricField(){return m_dMaxElectricField;}

        virtual int initialize( Geometry *geometry );

        //TM Functions
        virtual double getMagneticFieldZx( int i, int n );
        virtual double getMagneticFieldZy( int j, int n );
        virtual double getElectricFieldX( int i, int n );
        virtual double getElectricFieldY( int j, int n );

        virtual ~PulsedGaussianSource();

};

#endif
```

PulsedGaussianSource.cpp

```
#include "PulsedGaussianSource.h"
#include "GaussianSource.h"
#include <iostream.h>
#include <fstream.h>
#include <math.h>
#include "Geometry.h"


PulsedGaussianSource::PulsedGaussianSource(long index, char *name, Material *lastMaterial, double maxEField, int startTime, double lambda, double FWHM,
        double pulseFWHM, double phi) :GaussianSource( index, name, lastMaterial, maxEField, startTime, lambda, FWHM ){
        ifstream phase;
        phase.open("phase.txt");
        phase>>m_dPhi;
        m_dPhi *= PI/180.0;

        m_dPulseWidth = pulseFWHM;
        m_dPulseOffset = 2*m_dPulseWidth;
        m_dPulseSigma = m_dPulseWidth / ( sqrt( 2 * log( 2 ) ) );
        m_dChirp=0.0;//1e28;

        m_dMaxMagneticField=m_dMaxElectricField*sqrt( EPSILON_NOT
                / MU_NOT )*getRefractiveIndex();
}

int PulsedGaussianSource::initialize( Geometry *geometry ){
```

```
	int rc = GaussianSource::initialize( geometry );
	if( m_dPulseOffset >= geometry->getRunningTime() ){
		cerr << "Warning: Pulse maximum occurs after the last time step\n";
	}
	return rc;
}


//TM Functions
double PulsedGaussianSource::getMagneticFieldZx( int i, int n ){
	//this is for a horizontal source
	//if we're a vertical source then all Z Magnetic field components
	//in the x direction should not exist
	if( m_iMinX == m_iMaxX )
		return 0;
	m_dMaxMagneticField=m_dMaxElectricField*sqrt( EPSILON_NOT / MU_NOT )*getRefractiveIndex();

	double offset = m_dDeltaSpace * sqrt( ( i - m_iHalfX ) * ( i - m_iHalfX ) );

	double HField = m_dMaxMagneticField*exp(-0.5*offset*offset/m_dSigma/m_dSigma);
	double time =  (n-m_iStartTime) >= 0 ? (n-m_iStartTime)*m_dDeltaTime : 0;

	HField*= cos(m_dAngularFrequency*(time - m_dPulseOffset)+m_dChirp*(time - m_dPulseOffset)*(time - m_dPulseOffset)+m_dPhi);

	HField*= exp( - (time-m_dPulseOffset)*(time-m_dPulseOffset) / ( m_dPulseSigma * m_dPulseSigma ) );

	return -HField;
}

double PulsedGaussianSource::getMagneticFieldZy( int j, int n )
{
	//if we're a horizontal source then all Z Magnetic field components
	//in the Y direction should not exist
	if( m_iMinY == m_iMaxY )
		return 0;

	double offset = m_dDeltaSpace * sqrt( ( j - m_iHalfY ) * ( j - m_iHalfY ) );

	double HField = m_dMaxMagneticField*exp(-0.5*offset*offset/m_dSigma/m_dSigma);
	double time =  (n-m_iStartTime) >= 0 ? (n-m_iStartTime)*m_dDeltaTime : 0;

	HField*= cos(  (m_dAngularFrequency + m_dPhi*(time - m_dPulseOffset) )*(time - m_dPulseOffset) );

	HField*= exp( - (time-m_dPulseOffset)*(time-m_dPulseOffset)
		/ ( m_dPulseSigma * m_dPulseSigma ) );

	return -HField;
}

double PulsedGaussianSource::getElectricFieldX( int i, int n ){
	//this is for a horizontal source
	//if we're a vertical source then all X Electric field components
	//should not exist
	if( m_iMinX == m_iMaxX )
		return 0;

	double offset = m_dDeltaSpace * sqrt( ( i - m_iHalfX ) * ( i - m_iHalfX ) );
	double EField = m_dMaxElectricField*exp(-0.5*offset*offset/m_dSigma/m_dSigma);
	double time =  (n-m_iStartTime) > 0 ? (n-m_iStartTime+0.5)*m_dDeltaTime : 0;

	EField*= cos(m_dAngularFrequency*(time - m_dPulseOffset)+m_dChirp*(time - m_dPulseOffset)*(time - m_dPulseOffset)+m_dPhi);

	EField*= exp( - (time-m_dPulseOffset+INVERSE_LIGHT_SPEED*0.5*m_dDeltaSpace*getRefractiveIndex())*(time-m_dPulseOffset
		+INVERSE_LIGHT_SPEED*0.5*m_dDeltaSpace*getRefractiveIndex()) / ( m_dPulseSigma * m_dPulseSigma ) );

	return EField;
}

double PulsedGaussianSource::getElectricFieldY( int j, int n ){
	//if we're a horizontal source then all Y Electric field components
	//should not exist
	if( m_iMinY == m_iMaxY )
		return 0;

	double offset = m_dDeltaSpace * sqrt( ( j - m_iHalfY ) * ( j - m_iHalfY ) );
	double EField = m_dMaxElectricField*exp(-0.5*offset*offset/m_dSigma/m_dSigma);
	double time =  (n-m_iStartTime) > 0 ? (n-m_iStartTime+0.5)*m_dDeltaTime : 0;

	EField*= cos(  (m_dAngularFrequency + m_dPhi*(time - m_dPulseOffset) )*(time - m_dPulseOffset) );

	EField*= exp( - (time-m_dPulseOffset+INVERSE_LIGHT_SPEED*0.5*m_dDeltaSpace*getRefractiveIndex())*(time-m_dPulseOffset
		+INVERSE_LIGHT_SPEED*0.5*m_dDeltaSpace*getRefractiveIndex()) / ( m_dPulseSigma * m_dPulseSigma ) );

	return EField;
}

PulsedGaussianSource::~PulsedGaussianSource()
{
	cerr << "Message: Deleting pulsed gaussian source\n";
}
```

PulsedSecondOrderSolver.h

```
#ifndef PULSED_SECOND_ORDER_YEE_H
#define PULSED_SECOND_ORDER_YEE_H

#include "SecondOrderSolver.h"
#include "Geometry.h"

class PulsedSecondOrderSolver:public SecondOrderSolver
{
protected:
        //TM Mode    //TM Mode
        double** m_DElectricFieldPrevX;
        double** m_DElectricFieldPrevY;
        double** m_DCurrentDensityX;
        double** m_DCurrentDensityY;
        double** m_DElectricFieldDisplacementXPrevPrev;
        double** m_DElectricFieldDisplacementXPrev;
        double** m_DElectricFieldDisplacementX;
        double** m_DElectricFieldDisplacementYPrevPrev;
        double** m_DElectricFieldDisplacementYPrev;
        double** m_DElectricFieldDisplacementY;
        double** m_DLinearPolarization1XPrevPrev;
        double** m_DLinearPolarization1XPrev;
        double** m_DLinearPolarization1YPrevPrev;
        double** m_DLinearPolarization1YPrev;
        double** m_DLinearPolarization2XPrevPrev;
        double** m_DLinearPolarization2XPrev;
        double** m_DLinearPolarization2YPrevPrev;
        double** m_DLinearPolarization2YPrev;
        double** m_DLinearPolarization3XPrevPrev;
        double** m_DLinearPolarization3XPrev;
        double** m_DLinearPolarization3YPrevPrev;
        double** m_DLinearPolarization3YPrev;

        int initialize();
        int solveTMMode();

public:
        PulsedSecondOrderSolver( Geometry *geometry, Options *opt );
        virtual int solve();
        virtual ~PulsedSecondOrderSolver();
};
#endif
```

PulsedSecondOrderSolver.cpp

```
#include "PulsedSecondOrderSolver.h"
#include "Geometry.h"
#include "iostream.h"
#include "PMLBC.h"
#include "BitmapFieldWriter.h"
#include "Electron_Motion.h"
#include <math.h>
#include <cstdio>
#include <fstream.h>
#include "Electron_Group.h"
#include "Electron_Source_Finder.h"
#include "Electron_Source_Writer.h"

PulsedSecondOrderSolver::PulsedSecondOrderSolver( Geometry *geom, Options *options ) :SecondOrderSolver( geom, options ){
        cout <<"Creating: PulsedSecondOrderSolver\n";

        m_DElectricFieldX = NULL;
        m_DElectricFieldY = NULL;
        m_DMagneticFieldZ = NULL;
}

PulsedSecondOrderSolver::~PulsedSecondOrderSolver(){
        cout <<"Deleting: PulsedSecondOrderSolver\n";
        geometry = NULL;
}

int PulsedSecondOrderSolver::solve(){
        int rc = SUCCESS;

        if( options->getSimulationMode() == TM_MODE ){
                cerr << "Message: Solving geometry for TM Mode\n";
                rc = solveTMMode();
                if ( rc != SUCCESS ){ return rc; }
        }

        return rc;
}

int PulsedSecondOrderSolver::initialize(){
        int rc = FDTDSolver::initialize();
        if( rc != SUCCESS )
                return rc;
```

```
        //check the time and space deltas...
        cerr << "Message: Initializing Second Order Solver\n";
        if( geometry->getDeltaTime() > geometry->getDeltaSpace()/( LIGHT_SPEED )/sqrt(2.0) ){
                cerr << "Warning: Upper bound on lattice spacing is not met!\n";
                cerr << "Message: DeltaSpace = " << geometry->getDeltaSpace() << "\n";
                cerr << "Message: DeltaTime = " << geometry->getDeltaTime() << "\n";
                rc = NUMERICAL_STABILITY_WARNING;
        }
        return rc;
}

int PulsedSecondOrderSolver::solveTMMode(){
        int rc = SUCCESS;
        int i,j,n;
        int width = geometry->getGridWidth();
        int height = geometry->getGridHeight();
        int time = geometry->getGridTime();
        int ** electron_locations;
        double ** electron_numbers;

        bool electron_calculation=true;
        bool create_files=false*electron_calculation;
        int number_of_electrons_per_group=1250;
        int number_of_electron_groups=400;
        int x_extent=800;
        Electron_Group *electrons=new Electron_Group[number_of_electron_groups];
        ofstream number_of_electrons("number_of_electrons.txt");
        number_of_electrons<<number_of_electrons_per_group<<endl;
        number_of_electrons<<number_of_electron_groups<<endl;
        number_of_electrons.close();


        if(electron_calculation){
                Electron_Source_Finder electron_locator(x_extent,number_of_electron_groups,geometry);
                for(i=0;i<number_of_electron_groups;i++){
                        electrons[i].setCreateFiles(create_files);
                        electrons[i].SetInitialConditions(number_of_electrons_per_group,
                        i,electron_locator.get_x_array()[i],
                        electron_locator.get_y_array()[i],geometry);
                }
                Electron_Source_Writer e_writer(electron_locator.get_x_array(), electron_locator.get_y_array(),number_of_electron_groups,geometry);
        }


        double deltaTime = geometry->getDeltaTime();
        double deltaSpace = geometry->getDeltaSpace();

        PMLBC *bc = PMLBC::getBoundaryCondition();

        int PML_width=bc->getWidth();
        int PML_top = height - PML_width;
        int PML_bottom = PML_width-1;
        int PML_right = width - PML_width;
        int PML_left = PML_width-1;

        double h1,h2,h3,h4;
        long int oldIndex = -1;

        m_DMagneticFieldZ = new double*[ width ];
        m_DElectricFieldX = new double*[ width ];
        m_DElectricFieldY = new double*[ width ];
        m_DTotalElectricField = new double*[width];
        m_DElectricFieldPrevX = new double*[ width ];
        m_DElectricFieldPrevY = new double*[ width ];
        m_DMagneticFieldPrevZ = new double*[ width ];


        m_DElectricFieldDisplacementXPrevPrev = new double*[ width ];
        m_DElectricFieldDisplacementXPrev = new double*[ width ];
        m_DElectricFieldDisplacementX = new double*[ width ];
        m_DElectricFieldDisplacementYPrevPrev = new double*[ width ];
        m_DElectricFieldDisplacementYPrev = new double*[ width ];
        m_DElectricFieldDisplacementY = new double*[ width ];

        //extra arrays for boundary condition
        double **m_DMagneticFieldZX;
        double **m_DMagneticFieldZY;
        m_DMagneticFieldZX = new double*[ width ];
        m_DMagneticFieldZY = new double*[ width ];
        electron_locations = new int*[ width ];
        electron_numbers = new double*[ width ];


        //initialize the actual fields
        for( i=0; i<width; i++ ){
                m_DElectricFieldDisplacementXPrevPrev[i] = new double[ height ];
                m_DElectricFieldDisplacementXPrev[i] = new double[ height ];
                m_DElectricFieldDisplacementX[i] = new double[ height ];
                m_DElectricFieldDisplacementYPrevPrev[i] = new double[ height ];
                m_DElectricFieldDisplacementYPrev[i] = new double[ height ];
                m_DElectricFieldDisplacementY[i] = new double[ height ];

                m_DMagneticFieldZ[i] = new double [ height ];
```

```
                    m_DElectricFieldX[i] = new double [ height ];
                    m_DElectricFieldY[i] = new double [ height ];
                    m_DTotalElectricField[i] = new double [height];
                    m_DElectricFieldPrevX[i] = new double [ height ];
                    m_DElectricFieldPrevY[i] = new double [ height ];
                    m_DMagneticFieldPrevZ[i] = new double [ height ];

                    m_DMagneticFieldZX[i] = new double[ height ];
                    m_DMagneticFieldZY[i] = new double[ height ];

                    electron_locations[i] = new int [height];
                    electron_numbers[i] = new double [height];

                    for( j=0; j<height; j++ ){
                            m_DMagneticFieldZ[i][j] = 0; //add on the incident beam at t=0
                            m_DElectricFieldX[i][j] = 0;
                            m_DElectricFieldY[i][j] = 0;
                            m_DTotalElectricField [i][j] = 0;
                            m_DElectricFieldPrevX[i][j] = 0;
                            m_DElectricFieldPrevY[i][j] = 0;
                            m_DMagneticFieldPrevZ[i][j] = 0;

                            m_DElectricFieldDisplacementXPrevPrev[i][j] = 0;
                            m_DElectricFieldDisplacementXPrev[i][j] = 0;
                            m_DElectricFieldDisplacementX[i][j] = 0;

                            m_DElectricFieldDisplacementYPrevPrev[i][j] = 0;
                            m_DElectricFieldDisplacementYPrev[i][j] = 0;
                            m_DElectricFieldDisplacementY[i][j] = 0;

                            m_DMagneticFieldZX[i][j] = 0;
                            m_DMagneticFieldZY[i][j] = 0;

                            electron_locations[i][j]=0;
                            electron_numbers[i][j]=0;
                    }
            }

            //loop through the time steps...
            for( n=0; n<time; n++ )     {
                    //loop through the space region
                    cerr << "Message: " << double(n)/double(time)*100
                            << " percent complete\n";
                    for( i=0; i<width; i++ ){
                            for( j=0; j<height; j++ ){
                                    Material *mat = geometry->getMaterial(i,j);

                                    double tempEFieldX = m_DElectricFieldX[i][j];
                                    double tempEFieldY = m_DElectricFieldY[i][j];

if(j<PML_width||j>=height-PML_width||i<PML_width||i>=width-PML_width)
{
double      mAx = 0,mBx = 0,mAy = 0,mBy = 0;
double      eAx = 0,eBx = 0,eAy = 0,eBy = 0;

//were in the left layer
if( i < PML_width ){
        //were in the bottom left corner
        if( j < PML_width ){
                mAy = bc->getMagneticFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(),PML_width - j - 1);
                mBy = bc->getMagneticFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), PML_width - j - 1 );
                mAx = bc->getMagneticFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), PML_width - i - 1 );
                mBx = bc->getMagneticFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), PML_width - i - 1 );
                eAx = bc->getElectricFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), PML_width - j - 1 );
                eBx = bc->getElectricFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), PML_width - j - 1 );
                eAy = bc->getElectricFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), PML_width - i - 1 );
                eBy = bc->getElectricFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), PML_width - i - 1 );
        }
        //were in the top left corner
        else if( j >= height - PML_width ){
                mAx = bc->getMagneticFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), PML_width - i - 1 );
                mBx = bc->getMagneticFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), PML_width - i - 1 );
                mAy = bc->getMagneticFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), j-height+PML_width );
                mBy = bc->getMagneticFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), j-height+PML_width );
                eAy = bc->getElectricFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), PML_width - i - 1 );
                eBy = bc->getElectricFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), PML_width - i - 1 );
                eAx = bc->getElectricFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), j-height+PML_width );
                eBx = bc->getElectricFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), j-height+PML_width );
        }
        //were in the left layer only
        else{
                mAx = bc->getMagneticFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), PML_width - i - 1 );
                mBx = bc->getMagneticFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), PML_width - i - 1 );
                mAy = bc->getMagneticCoefficientA(geometry,mat->getElectricPermittivity());
                mBy = bc->getMagneticCoefficientB(geometry,mat->getElectricPermittivity());
                eAy = bc->getElectricFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), PML_width - i - 1 );
                eBy = bc->getElectricFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), PML_width - i - 1 );
                eAx = bc->getElectricCoefficientA(geometry,mat->getElectricPermittivity());
                eBx = bc->getElectricCoefficientB(geometry,mat->getElectricPermittivity());
        }
}
else if( i >= width - PML_width ){
        //were in the bottom right corner
```

```
		if( j < PML_width ){
			mAx = bc->getMagneticFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), i-width+PML_width );
			mBx = bc->getMagneticFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), i-width+PML_width );
			mAy = bc->getMagneticFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), PML_width - j - 1 );
			mBy = bc->getMagneticFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), PML_width - j - 1 );
			eAy = bc->getElectricFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), i-width+PML_width );
			eBy = bc->getElectricFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), i-width+PML_width );
			eAx = bc->getElectricFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), PML_width - j - 1 );
			eBx = bc->getElectricFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), PML_width - j - 1 );
		}
		//were in the top right corner
		else if( j >= height - PML_width ){
			mAx = bc->getMagneticFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), i-width+PML_width );
			mBx = bc->getMagneticFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), i-width+PML_width );
			mAy = bc->getMagneticFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), j-height+PML_width );
			mBy = bc->getMagneticFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), j-height+PML_width );
			eAy = bc->getElectricFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), i-width+PML_width );
			eBy = bc->getElectricFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), i-width+PML_width );
			eAx = bc->getElectricFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), j-height+PML_width );
			eBx = bc->getElectricFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), j-height+PML_width );
		}
		//were in the right layer only
		else{
			mAx = bc->getMagneticFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), i-width+PML_width );
			mBx = bc->getMagneticFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), i-width+PML_width );
			mAy = bc->getMagneticCoefficientA(geometry,mat->getElectricPermittivity());
			mBy = bc->getMagneticCoefficientB(geometry,mat->getElectricPermittivity());
			eAy = bc->getElectricFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), i-width+PML_width );
			eBy = bc->getElectricFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), i-width+PML_width );
			eAx = bc->getElectricCoefficientA(geometry,mat->getElectricPermittivity());
			eBx = bc->getElectricCoefficientB(geometry,mat->getElectricPermittivity());
		}
	}
	//were in the bottom layer only
	else if( j < PML_width ){
		mAx = bc->getMagneticCoefficientA(geometry,mat->getElectricPermittivity());
		mBx = bc->getMagneticCoefficientB(geometry,mat->getElectricPermittivity());
		mAy = bc->getMagneticFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), PML_width - j - 1 );
		mBy = bc->getMagneticFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), PML_width - j - 1 );
		eAy = bc->getElectricCoefficientA(geometry,mat->getElectricPermittivity());
		eBy = bc->getElectricCoefficientB(geometry,mat->getElectricPermittivity());
		eAx = bc->getElectricFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), PML_width - j - 1 );
		eBx = bc->getElectricFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), PML_width - j - 1 );
	}
	//were in the top layer only
	else if( j >= height - PML_width ){
		mAx = bc->getMagneticCoefficientA(geometry,mat->getElectricPermittivity());
		mBx = bc->getMagneticCoefficientB(geometry,mat->getElectricPermittivity());
		mAy = bc->getMagneticFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), j-height+PML_width );
		mBy = bc->getMagneticFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), j-height+PML_width );
		eAy = bc->getElectricCoefficientA(geometry,mat->getElectricPermittivity());
		eBy = bc->getElectricCoefficientB(geometry,mat->getElectricPermittivity());
		eAx = bc->getElectricFieldExponentialCoefficientA(geometry,mat->getElectricPermittivity(), j-height+PML_width );
		eBx = bc->getElectricFieldExponentialCoefficientB(geometry,mat->getElectricPermittivity(), j-height+PML_width );
	}
	else{
		cerr << "Error: Enterened erroneous region!\n";
	}

                                                double Hzx = j==height-1?0.0:m_DMagneticFieldZ[i][j+1];
                                                double Hzy = i==width-1?0.0:m_DMagneticFieldZ[i+1][j];

                                                m_DElectricFieldX[i][j]=eAx*m_DElectricFieldX[i][j]- eBx*(m_DMagneticFieldZ[i][j]-Hzx);
                                                m_DElectricFieldY[i][j]=eAy*m_DElectricFieldY[i][j]-eBy*(Hzy-m_DMagneticFieldZ[i][j]);

                                                double Ey = i==0?0.0:m_DElectricFieldY[i-1][j];
                                                double Ex = j==0?0.0:m_DElectricFieldX[i][j-1];

                                                m_DMagneticFieldZX[i][j]=mAx*m_DMagneticFieldZX[i][j]-mBx*(m_DElectricFieldY[i][j]-Ey);

                                                m_DMagneticFieldZY[i][j]=mAy*m_DMagneticFieldZY[i][j]-mBy*(-m_DElectricFieldX[i][j]+Ex);
                                                m_DMagneticFieldPrevZ[i][j]=m_DMagneticFieldZ[i][j];
                                                m_DMagneticFieldZ[i][j]=m_DMagneticFieldZX[i][j]+m_DMagneticFieldZY[i][j];
                                        }
                                        else{
                                                double A = mat->getElectricCoefficientA();
                                                double B = mat->getElectricCoefficientB();
                                                double Bo = geometry->getDeltaTime()/geometry->getDeltaSpace();

                                                double Hzx = j==height-1?0.0:m_DMagneticFieldZ[i][j+1];
                                                double Hzy = i==width-1?0.0:m_DMagneticFieldZ[i+1][j];

                                                double HzIncX = 0;
                                                double HzIncY = 0;

                                                Source *source = NULL;

                                                if( ( source = geometry->isPointLeftOfSource( i, j ) ) != NULL ){
                                                        HzIncY = source->getMagneticFieldZy( j, n );

                                                }
                                                else if( ( source = geometry->isPointBelowSource( i, j ) ) != NULL ){
                                                        HzIncX = source->getMagneticFieldZx( i, n );
```

```
                                        }

                                        m_DElectricFieldDisplacementX[i][j]=m_DElectricFieldDisplacementX[i][j]
                                                +Bo*(Hzx-m_DMagneticFieldZ[i][j]-HzIncX);
                                        m_DElectricFieldDisplacementY[i][j] = m_DElectricFieldDisplacementY[i][j]
                                                +Bo*(m_DMagneticFieldZ[i][j]-Hzy+HzIncY);

                                        if( mat->isDispersive() ){
                                                if( mat->isMetal() ){//Drude Model
                                                        h1 = mat->getCoefficientH1();
                                                        h2 = mat->getCoefficientH2();
                                                        h3 = mat->getCoefficientH3();
                                                        h4 = mat->getCoefficientH4();

                                                        m_DElectricFieldX[i][j] = ( h1*m_DElectricFieldDisplacementX[i][j]
                                                                - 4*m_DElectricFieldDisplacementXPrev[i][j]
                                                                + h2*m_DElectricFieldDisplacementXPrevPrev[i][j]
                                                                + 4*mat-getElectricPermittivity()*m_DElectricFieldX[i][j]
                                                                - h3*m_DElectricFieldPrevX[i][j] )
                                                                / ( h4 );

                                                        m_DElectricFieldY[i][j] = ( h1*m_DElectricFieldDisplacementY[i][j]
                                                                - 4*m_DElectricFieldDisplacementYPrev[i][j]
                                                                + h2*m_DElectricFieldDisplacementYPrevPrev[i][j]
                                                                + 4*mat-getElectricPermittivity()*m_DElectricFieldY[i][j]
                                                                - h3*m_DElectricFieldPrevY[i][j] )
                                                                / ( h4 );

                                                }
                                        }
                                        else                                                    {
                                                A = mat->getElectricCoefficientA();
                                                B = mat->getElectricCoefficientB();
                                                m_DElectricFieldX[i][j] = A * m_DElectricFieldX[i][j]
                                                        + B * ( Hzx
                                                        - m_DMagneticFieldZ[i][j]
                                                        - HzIncX);

                                                m_DElectricFieldY[i][j] = A * m_DElectricFieldY[i][j]
                                                        + B * ( m_DMagneticFieldZ[i][j]
                                                        - Hzy
                                                        + HzIncY);
                                        }

                                        m_DElectricFieldPrevX[i][j] = tempEFieldX;
                                        m_DElectricFieldPrevY[i][j] = tempEFieldY;
                                        m_DElectricFieldDisplacementXPrevPrev[i][j] = m_DElectricFieldDisplacementXPrev[i][j];
                                        m_DElectricFieldDisplacementXPrev[i][j] = m_DElectricFieldDisplacementX[i][j];
                                        m_DElectricFieldDisplacementYPrevPrev[i][j] = m_DElectricFieldDisplacementYPrev[i][j];
                                        m_DElectricFieldDisplacementYPrev[i][j] = m_DElectricFieldDisplacementY[i][j];

                                        Material *mat = geometry->getMaterial(i,j);

                                        double Ey = i==0?0.0:m_DElectricFieldY[i-1][j];
                                        double Ex = j==0?0.0:m_DElectricFieldX[i][j-1];

                                        A = mat->getMagneticCoefficientA();
                                        B = mat->getMagneticCoefficientB();

                                        double EIncX = 0;
                                        double EIncY = 0;

                                        if( mat->isSource() ){
                                                Source *source = (Source *) mat;
                                                EIncX = source->getElectricFieldX( i, n );
                                                EIncY = source->getElectricFieldY( j, n );
                                        }

                                        m_DMagneticFieldPrevZ[i][j]=m_DMagneticFieldZ[i][j];
                                        m_DMagneticFieldZ[i][j]=A*m_DMagneticFieldZ[i][j]
                                                +B*(m_DElectricFieldX[i][j]-Ex-m_DElectricFieldY[i][j]
                                                +Ey-EIncX+EIncY);
                                }
                        // calculate the total electric field
                        m_DTotalElectricField[i][j] = sqrt(m_DElectricFieldY[i][j]*m_DElectricFieldY[i][j] +
                                m_DElectricFieldX[i][j]*m_DElectricFieldX[i][j]);
                        } // i loop
} //j loop

//forward iterate the electrons
if(electron_calculation){

        for(int l=0;l<width;l++){
                for(int m=0;m<height;m++){
                        electron_locations[l][m]=0;
                        electron_numbers[l][m]=0;
                }
        }
        for(i=0;i<number_of_electron_groups;i++){
                electrons[i].Forward_Iterate(m_DElectricFieldX,m_DElectricFieldY,m_DMagneticFieldZ,
                        m_DElectricFieldPrevX,m_DElectricFieldPrevY,m_DMagneticFieldPrevZ,geometry,n);
                for(j=0;j<number_of_electrons_per_group;j++){
                        int i_location=electrons[i].getElectron(j).getX()/deltaSpace;
```

```
                                                int j_location=electrons[i].getElectron(j).getY()/deltaSpace;

                                                if(electrons[i].getElectron(j).isRecaptured()){
                                                }
                                                else{
                                                        if(electrons[i].getElectron(j).isActive()){
                                                                if((i_location>=0)&&(i_location<width)
                                                                        &&(j_location>=0)&&(j_location<height)){
                                                                electron_locations[i_location][j_location]=1;
                                                                electron_numbers[i_location][j_location]
                                                                        +=(-1.0)
                                                                        *electrons[i].getElectron(j).getNumber();
                                                                }
                                                        }
                                                }
                                        }
                                }

                                //output a field veetor
                                if( options->getFrameSpacing() != 0 && (n+1)%options->getFrameSpacing() == 0 ){
                                        char filename[512];
                                        sprintf(filename,"%s_%s_%s_%4d",options->getSimulationName(),  TM_MODE_TOKEN,"Et",(n+1)/options->getFrameSpacing());
                                        rc = BitmapFieldWriter::write( filename, m_DTotalElectricField, width, height, options,electron_locations,electron_numbers);
                                }

                } // n loop

//write the electron information
                if(electron_calculation){
                        ofstream final_velocities;
                        final_velocities.open("final_electron_velocities.txt");
                        for( i=0; i<number_of_electron_groups; i++ ){
                                for( j=0; j<number_of_electrons_per_group; j++ ){
                                        final_velocities
                                                <<electrons[i].getElectron(j).getX()<<" "
                                                <<electrons[i].getElectron(j).getY()<<" "
                                                <<electrons[i].getElectron(j).getVx()<<" "
                                                <<electrons[i].getElectron(j).getVy()<<" "
                                                <<electrons[i].getElectron(j).getNumber()<<" "
                                                <<electrons[i].getElectron(j).isRecaptured()<<endl;
                                }
                        }
                        final_velocities.close();
                }

                //delete the arrays
                for( i=0; i<width; i++ ){
                        delete [] m_DTotalElectricField[i];
                        delete [] m_DMagneticFieldZ[i];
                        delete [] m_DMagneticFieldPrevZ[i];
                        delete [] m_DElectricFieldX[i];
                        delete [] m_DElectricFieldY[i];
                        delete [] m_DMagneticFieldZX [i];
                        delete [] m_DMagneticFieldZY [i];

                }
                delete [] m_DTotalElectricField;
                delete [] m_DMagneticFieldZ;
                delete [] m_DMagneticFieldPrevZ;
                delete [] m_DElectricFieldX;
                delete [] m_DElectricFieldY;
                delete [] m_DMagneticFieldZX;
                delete [] m_DMagneticFieldZY;
                cerr << "Message: Finished TM Mode\n\n";
                return rc;
}
```

SecondOrderSolver.h

```
#ifndef SECOND_ORDER_YEE_H
#define SECOND_ORDER_YEE_H

#include "FDTDSolver.h"
#include "Geometry.h"

class SecondOrderSolver:public FDTDSolver
{
protected:
                //TM Mode
                double** m_DElectricFieldX;
                double** m_DElectricFieldY;
                double** m_DMagneticFieldZ;
                double** m_DMagneticFieldPrevZ;

                //TE Mode
                double** m_DMagneticFieldX;
                double** m_DMagneticFieldY;
                double** m_DElectricFieldZ;
```

```
        double** m_DTotalElectricField;


public:
        SecondOrderSolver( Geometry *geometry, Options *opt );
        virtual ~SecondOrderSolver();
};
#endif
```

SecondOrderSolver.cpp

```
#include "SecondOrderSolver.h"
#include "Geometry.h"
#include "iostream.h"
#include "FDTDSolver.h"
#include "PMLBC.h"
#include "GaussianSource.h"
#include "BitmapFieldWriter.h"
#include <math.h>
#include <cstdio>

SecondOrderSolver::SecondOrderSolver( Geometry *geom, Options *options ):FDTDSolver( geom, options ){
        cout <<"Creating: SecondOrderSolver\n";
        //TM Mode
        m_DElectricFieldX = NULL;
        m_DElectricFieldY = NULL;
        m_DMagneticFieldZ = NULL;

        m_DTotalElectricField = NULL;
}

SecondOrderSolver::~SecondOrderSolver(){
        cout <<"Deleting: SecondOrderSolver\n";
        geometry = NULL;
}
```

Source.h

```
#ifndef SOURCE_H
#define SOURCE_H

class Source : public Material{
protected:
        double m_dLambda;
        double m_dMaxElectricField;
        double m_dDeltaSpace;
        double m_dDeltaTime;
        double m_dMaxMagneticField;
        double m_dAngularFrequency;
        double m_dFrequency;

        double m_iHalfX;
        double m_iHalfY;
        int m_iStartTime;

        int m_iMaxX;
        int m_iMaxY;
        int m_iMinX;
        int m_iMinY;

public:
        Source( long index, char *name, Material *lastMaterial,double maxEField, int startTime, double wavelength);

        virtual int initialize( Geometry *geometry );
        virtual double getElectricFieldZx( int i, int n );
        virtual double getElectricFieldZy( int j, int n );
        virtual double getMagneticFieldX( int i, int n );
        virtual double getMagneticFieldY( int j, int n );

        virtual double getMagneticFieldZx( int i, int n );
        virtual double getMagneticFieldZy( int j, int n );
        virtual double getElectricFieldX( int i, int n );
        virtual double getElectricFieldY( int j, int n );
        void setPoint( int x, int y);

        virtual ~Source();
};

#endif
```

Source.cpp

```
#include "GaussianSource.h"
#include <iostream.h>
#include <math.h>
#include "stdheader.h"
#include "Geometry.h"
```

```
#include <stdio.h>
#include "PulsedGaussianSource.h"

Source::Source( long index, char *name, Material *lastMaterial, double maxEField, int startTime, double wavelength )
:Material( index, name, lastMaterial){
        m_dLambda = wavelength;
        m_dFrequency = LIGHT_SPEED / m_dLambda;
        m_dAngularFrequency = 2 * PI * m_dFrequency;
        m_blsSource = true;
        m_dMaxElectricField = maxEField;
        m_dMaxMagneticField = maxEField*sqrt( getElectricPermittivity() / getMagneticPermeability() );
        m_iStartTime = startTime;
        m_iMinX = MAX_INT;
        m_iMinY = MAX_INT;
        m_iMaxX = MIN_INT;
        m_iMaxY = MIN_INT;

        cerr << "Message: Creating source\n";
}

int Source::initialize( Geometry *geometry ){
        int rc = Material::initialize( geometry );
        if( rc != SUCCESS )
                return rc;

        geometry->setMinimumLambda( m_dLambda );
        m_dDeltaSpace = geometry->getDeltaSpace();
        m_dDeltaTime = geometry->getDeltaTime();

        int dy = (m_iMaxY - m_iMinY);
        int dx = (m_iMaxX - m_iMinX);

        m_iHalfX = ( dx/2 + m_iMinX );
        m_iHalfY = ( dy/2 + m_iMinY );

        if( m_iStartTime > geometry->getRunningTime() )
                cerr << "Warning: Gaussian source is non-propagating due to time constraints\n";
        return rc;
}

void Source::setPoint( int x, int y){
        if( x < m_iMinX )
                m_iMinX = x;
        if( x > m_iMaxX )
                m_iMaxX = x;
        if( y < m_iMinY )
                m_iMinY = y;
        if( y > m_iMaxY )
                m_iMaxY = y;
}

double Source::getElectricFieldZx( int i, int n ){return 0;}
double Source::getElectricFieldZy( int j, int n ){return 0;}
double Source::getMagneticFieldX( int i, int n ){return 0;}
double Source::getMagneticFieldY( int j, int n ){return 0;}
double Source::getMagneticFieldZx( int i, int n){return 0;}
double Source::getMagneticFieldZy( int j, int n ){return 0;}
double Source::getElectricFieldX( int i, int n ){return 0;}
double Source::getElectricFieldY( int j, int n ){return 0;}
Source::~Source(){cerr << "Message: Deleting source\n";}
```

stdheader.h

```
#ifndef STD_HEADER_H
#define STD_HEADER_H

#define EPSILON_NOT 8.8542E-12
#define INVERSE_EPSILON_NOT 1.0/8.8542E-12
#define MU_NOT 1.2566E-6
#define INVERSE_MU_NOT 1.0/1.2566E-6
#define CHARGE_MASS_RATIO (-1.758820174e11)
#define LIGHT_SPEED 3.0E8
#define INVERSE_LIGHT_SPEED 1.0/LIGHT_SPEED
#define PI 3.14159265358979323846264338327 95

#define SUCCESS 0
#define FILE_NOT_FOUND 1
#define INVALID_FILE_FORMAT 2
#define NUMERICAL_STABILITY_WARNING 3
#define TIME_STEP_WARNING 4
#define INVALID_FILENAME 5

#define MAX_INT 1000000000
#define MIN_INT -1000000000

#define BIT_DEPTH 24
#define BITS_PER_BYTE 8

#ifndef BYTE_I
#define BYTE_I unsigned char
#endif
```

```
#define GRID_TOKEN "[GRID]"
#define MATERIAL_TOKEN "[MATERIALS]"
#define OPTIONS_TOKEN "[OPTIONS]"

#define DELTA_TOKEN "DELTA="
#define DELTA_TIME_TOKEN "DELTA_TIME="
#define LAMBDA_TOKEN "LAMBDA="
#define GRID_TIME_TOKEN "GRID_TIME="
#define FRAME_SPACING_TOKEN "FRAME_SPACING="
#define SIMULATION_NAME_TOKEN "SIMULATION_NAME="
#define SIMULATION_MODE_TOKEN "SIMULATION_MODE="
#define SIMULATION_OUTPUT_TOKEN "SIMULATION_OUTPUT="
#define TIME_DOMAIN_FIELD_OUTPUT_TOKEN "TIME_DOMAIN_FIELD_OUTPUT="
#define X_FIELD_INTEGRATOR_TOKEN "X_FIELD_INTEGRATOR="
#define Y_FIELD_INTEGRATOR_TOKEN "Y_FIELD_INTEGRATOR="
#define QUERY_CONTINUE_TOKEN "QUERY_CONTINUE="

#define MATERIAL_TYPE_TOKEN "MATERIAL"
#define DISPERSIVE_MATERIAL_TYPE_TOKEN "DISPERSIVE_MATERIAL"
#define METAL_MATERIAL_TYPE_TOKEN "METAL_MATERIAL"
#define CONDUCTIVE_MATERIAL_TYPE_TOKEN "CONDUCTIVE_MATERIAL"
#define BOUNDARY_TYPE_TOKEN "BOUNDARY"
#define SOURCE_TYPE_TOKEN "SOURCE"
#define GAUSSIAN_SOURCE_TOKEN "GAUSSIAN"
#define PULSED_GAUSSIAN_SOURCE_TOKEN "PULSED_GAUSSIAN"
#define PLANE_WAVE_SOURCE_TOKEN "PLANE_WAVE"
#define PERFECTLY_MATCHED_LAYER_TOKEN "PERFECTLY_MATCHED_LAYER"

#define NO_STATE 0
#define GRID_STATE 1
#define MATERIAL_STATE 2
#define OPTIONS_STATE 3

#define TM_TE_MODE 0
#define TE_MODE 1
#define TM_MODE 2
#define TM_MODE_TOKEN "TM_MODE"
#define TE_MODE_TOKEN "TE_MODE"
#define TM_TE_MODE_TOKEN "TM_TE_MODE"

#define BMP_OUTPUT_MODE_TOKEN "BMP"
#define TXT_OUTPUT_MODE_TOKEN "TXT"
#define BMP_TXT_OUTPUT_MODE_TOKEN "BMP_TXT"
#define BMP_OUTPUT_MODE 0
#define TXT_OUTPUT_MODE 1
#define BMP_TXT_OUTPUT_MODE 2

#endif
```

Curriculum Vitae

Education

Institution	From – To (yr/mo)	Name of Degree	Degree Conferred
University of Alberta	2002/09-Present	Ph.D. (Electrical Engineering)	Expected 2007/06
University of Alberta	2000/09-2002/09	M.Sc. (Electrical Engineering)	Yes
University of Lethbridge	1996/09-2000/04	B.Sc. (Math and Physics)	Yes

Research Experience

Position held (research area)	Organization	Supervisor	Period (year/month)
Visiting Scientist *(Few-cycle laser pulses)*	Max-Planck Institute of Quantum Optics, Garching, Germany	Dr. Ferenc Krausz	2005/11-2005/12
Ph.D. Graduate Researcher *(Femtosecond Electron Pulse Generation)*	Electrical Engineering, University of Alberta	Dr. Abdulhakem Elezzabi	2002/09-2006/10
M.Sc. Graduate Researcher *(Magneto-Optic Modulation)*	Electrical Engineering, University of Alberta	Dr. Abdulhakem Elezzabi	2000/09-2002/09
NSERC Undergraduate Research Student *(Fourier Transform Spectroscopy)*	Department of Physics, University of Lethbridge	Dr. David Naylor	2000/05-2000/08
NSERC Undergraduate Research Student *(Conformal Field Theory)*	Department of Physics, University of Lethbridge	Dr. Mark Walton	1999/05-1999/08
NSERC Undergraduate Research Student *(Nuclear Magnetic Resonance)*	Department of Physics, University of Lethbridge	Dr. David Siminovitch	1998/05-1998/08

Academic Awards

Name of Scholarship or Award	Period Held	Institution
Andrew Stewart Memorial Graduate Prize	2005/09-2006/09	University of Alberta
Izaak Walton Killam Memorial Scholarship	2005/05-2006/12	University of Alberta
Ralph Steinhauer Award of Distinction	2005/09-2006/09	University of Alberta
OSA Student Travel Award (International)	2004/07-2004/07	University of Alberta
Alberta Ingenuity Graduate Award	2003/05-2008/05	University of Alberta
NSERC PGS B Graduate Scholarship	2002/09-2004/09	University of Alberta
Alberta Learning Graduate Scholarship	2002/09-2003/09	University of Alberta
Walter H. Johns Scholarship	2000/09-2004/09	University of Alberta
iCore Scholarship	2000/09-2006/09	University of Alberta
NSERC PGS A Graduate Scholarship	2000/09-2002/09	University of Alberta
NSERC Summer Research Award	2000/05-2000/08	University of Lethbridge
George Ellis Scholarship	1999/09-2000/04	University of Lethbridge
NSERC Summer Research Award	1999/05-1999/08	University of Lethbridge
U of L Scholarship	1998/09-2000/04	University of Lethbridge
NSERC Summer Research Award	1998/05-1998/08	University of Lethbridge
Rogers Sugar Award	1997/09-1998/04	University of Lethbridge
Palliser Distillers Scholarship	1996/09-1997/04	University of Lethbridge
U of L Entrance Scholarship	1996/09-1997/04	University of Lethbridge
Alexander Rutherford Scholarship	1996/09-1997/04	University of Lethbridge

Refereed Publications

1) **S. E. Irvine**, P. Dombi, Gy. Farkas, and A. Y. Elezzabi, "Influence of Carrier-Envelope Phase of Few-Cycle Pulses on Surface-Plasmon Ponderomotive Electron Interaction," *Physical Review Letters*, vol. 97, pp. 146801, 2006.

2) **S. E. Irvine** and A. Y. Elezzabi, "Ultracompact 180° magnetic spectrometer for intermediate energy electron measurement," *Measurement Science and Technology*, vol. 17, pp. 2455-2460, 2006.

3) **S. E. Irvine** and A. Y. Elezzabi, "Femtosecond electron pulse gating using surface plasmons," *Optics Express*, vol. 14, pp. 4115-4127, 2006.

4) **S. E. Irvine** and A. Y. Elezzabi, "Surface-plasmon-based electron acceleration," *Physical Review A*, vol. 73, pp. 013815, 2006.

5) **S. E. Irvine** and A. Y. Elezzabi, "Ponderomotively Accelerated 2 keV Femtosecond Electron Packets using Surface Plasmon Waves," *Applied Physics Letters*, vol. 86, pp. 264102, 2005.

6) **S. E. Irvine**, A. Dechant, and A. Y. Elezzabi, "Generation of 0.4 keV Femtosecond Electron Pulses using Impulsively Excited Surface Plasmons," *Physical Review Letters*, vol. 93, pp. 184801, 2004.

7) **S. E. Irvine** and A. Y. Elezzabi, "Surface plasmon assisted 26 fs, 0.4 keV electron pulse generation," Niigata, Japan, in *Proceedings of 14th International Conference on Ultrafast Phenomena*, (Springer Series in Chemical Physics, New York), 2004.

8) A. Dechant, S. K. Dew, **S. E. Irvine**, and A. Y. Elezzabi, "High Transmission Solid Immersion Apertured Optical Probes for Near-Field Scanning Optical Microscopy," *Applied Physics Letters*, vol. 86, 013102, 2004.

9) K. J. Chau, **S. E. Irvine**, and A. Y. Elezzabi, "A Gigahertz Surface Magneto-Plasmon Optical Modulator," *IEEE Journal of Quantum Electronics*, vol. 40, pp. 571-579, 2004.

10) **S. E. Irvine** and A. Y. Elezzabi, "Polarization-Selective Optical Beam Deflection using Magnetically Activated Bi-YIG Films," *Proceedings of SPIE*, vol. 5260, pp. 585-590, 2003.

11) **S. E. Irvine** and A. Y. Elezzabi, "Broadband Optical Modulation using Magneto-Optic Bi-YIG Thin Films," *Proceedings of SPIE*, vol. 5260, pp. 580-584, 2003.

12) **S. E. Irvine** and A. Y. Elezzabi, "A Miniature Broadband Bismuth-Substituted Yttrium Iron Garnet Magneto-Optic Modulator," *Journal of Physics D: Applied Physics, Special issue on the interface between magnetism and photonics*, vol. 36, pp. 2218-2221, 2003.

13) **S. E. Irvine** and A. Y. Elezzabi, "Wideband magneto-optic modulation in a bismuth-substituted yttrium iron garnet waveguide," *Optics Communications*, vol. 220, pp. 325-329, 2003.

14) A. Y. Elezzabi and **S. E. Irvine**, "Parametrically driven first-order Suhl instability and nonlinearities in bismuth-substituted yttrium iron garnet films," *Applied Physics Letters*, vol. 82, pp. 2464-2466, 2003.

15) J. F. Holzman, F. E. Vermeulen, **S. E. Irvine**, and A. Y. Elezzabi, "Free-space detection of terahertz radiation using crystalline and polycrystalline ZnSe electro-optic sensors," *Applied Physics Letters*, vol. 81, pp. 2294-2296, 2002.

16) **S. E. Irvine** and A. Y. Elezzabi, "Modeling of High-Speed Magneto-Optic Beam Deflection," *IEEE Journal of Quantum Electronics*, vol. 38, pp. 1428-1435, 2002.

17) **S. E. Irvine** and A. Y. Elezzabi, "Multi-Gigahertz Guided Wave Magneto-Optic Modulator," *IEEE Photonics Technology Letters*, vol. 14, pp. 798-800, 2002.

18) W. Ying, **S. E. Irvine**, R. A. Beekman, D. J. Siminovitch, and S. O. Smith, "Deuterium NMR Reveals Helix Packing Interactions in Phospholamban," *Journal of the American Chemical Society*, vol. 122, pp. 11125-11128, 2000.

19) **S. E. Irvine** and M. A. Walton, "Schubert calculus and threshold polynomials of affine fusion," *Nuclear Physics B*, vol. 584, pp. 795-809, 2000.

Presentations

1) **S. E. Irvine**, "Surface Plasmon-Based Electron Acceleration (Part II)," Invited Colloquium, Department of Physics, University of Lethbridge, 2006.

2) **S. E. Irvine**, "Surface Plasmon-Based Electron Acceleration," *Max Planck Institute for Quantum Optics*, Garching, Germany, 2005.

3) **S. E. Irvine**, "Surface Plasmon-Based Electron Acceleration," *Max Planck Institute for Biophysical Chemistry*, Göttingen, Germany, 2005.

4) **S. E. Irvine** and A. Y. Elezzabi, "Generation of 2 keV, 30 fs Electron Pulses via Surface Plasmon Waves," *Conference on Lasers and Electro-Optics*, Baltimore, Maryland, 2005.

5) **S. E. Irvine** and A. Y. Elezzabi, "Experimental Investigation and Modeling of Electron Pulse Generation using Surface Plasmons", *American Physical Society March Meeting*, Los Angeles, California, 2005.

6) **S. E. Irvine**, "Laser-Field Femtosecond Electron Pulse Generation and Acceleration using Surface Plasmon Waves," Invited Colloquium, Department of Physics, University of Lethbridge, 2005.

7) **S. E. Irvine** and A. Y. Elezzabi, "Surface plasmon assisted 26 fs, 0.4 keV electron pulse generation," *International Conference on Ultrafast Phenomena*, Niigata, Japan, 2004.

8) **S. E. Irvine** and A. Y. Elezzabi, "Parametric Excitation of Spin Waves in Bismuth-substituted Yttrium Iron Garnet Films using the First-order Suhl Instability," *International Conference on MEMS, NANO, and Smart Systems*, Banff, Alberta, Canada, 2003.

9) **S. E. Irvine** and A. Y. Elezzabi, "Polarization-Selective Optical Beam Deflection using Magnetically Activated Bi-YIG Films," *Photonics North*, Montreal, Quebec, Canada, 2003.

10) **S. E. Irvine** and A. Y. Elezzabi, "Broadband Optical Modulation using Magneto-Optic Bi-YIG Thin Films," *Photonics North*, Montreal, Quebec, Canada, 2003.

Patent

1) U.S. Patent Pending, "High-Speed Magneto-Optic Modulator" (U.S. Provisional Patent No. 60/419,112)

Administrative/Teaching Experience

Oct. 2004-Present	SPIE Student Chapter Member and Co-founder Duties: Establishment of the Chapter's Constitution and Executive Members
Sept. 2003-Present	Department General Meetings, University of Alberta Duties: Student Representation
Jan. 2006	EE571 Optical and Quantum Electronics Duties: Guest Lectures on laser resonator design and electron acceleration
Sept. 2005-April 2006	EE350 Analog Electronics, Laboratory Instructor EE471 Photonics I, Laboratory Instructor Duties: Laboratory Instruction and Guidance
Sept. 2004-April 2005	EE452 Physical Electronics, Teaching Assistant EE474 Introduction to Plasmas, Teaching Assistant Duties: Grading assignments
Jan. 2004-July 2004	ECE Chairman Selection Committee, University of Alberta Duties: Selecting and Interviewing Candidates for Department Chair
Sept. 1999-Aug.2000	Department of Physics Chairman Selection Committee, University of Lethbridge Duties: Undergraduate Student Representation

Quantitative Skills

Experimental: Advanced ultrafast laser/amplifier operations, advanced optical experimentation (polarization, waveguide optics, fiber optics, etc.), nanofabrication processes (sputtering, deposition, photolithography, etc.), cryogenic material handling, high-vacuum system design and construction

Simulation: finite-difference time-domain, test-particle code, beam propagation methods

Computer Languages: Visual C++, FORTRAN, Maplé V, Java, MATLAB

Interests/Hobbies/Achievements

Volunteer Judge, Edmonton Regional Science Fair (2005-2006)

Rock Climbing (Member of the Alpine Club of Canada)

Weightlifting (1st place in teen category, Western Canadian Powerlifting Championship, 1995)

Violin/Fiddle